Improving Statistical Methods for RNA Sequencing Data: Outliers, Boundary Conditions, Benchmarking and Differential Transcript Expression

---

Dissertation

zur

Erlangung der naturwissenschaftlichen Doktorwürde
(Dr. sc. nat.)

vorgelegt der

Mathematisch-naturwissenschaftlichen Fakultät

der

Universität Zürich

von

Xiaobei Zhou

aus der

V.R. China

Promotionskommission

Prof. Dr. Mark D. Robinson (Vorsitz)
Prof. Dr. Torsten Hothorn
Prof. Dr. Reinhard Furrer
Prof. Dr. Christian von Mering

Zürich, 2017

Preface

I would like to thank all my committee members, including Prof. Mark D. Robinson, Prof. Torsten Hothorn, Prof. Reinhard Furrer and Prof. Christian von Mering, for their advice during my PhD studies. I also thank Prof. Lieven Clement for his instructive suggestions.

I give my most sincere thanks to Mark D. Robinson who has served as the supervisor for all the research topics described in this dissertation. I was so lucky to meet him when he founded his research group in Zurich and then became his first PhD student. I am grateful for his patience, hard work, open mind and optimal organization. He gave me the opportunity to enter the science world!

Moreover, I express my thanks to all members of the Robinson laboratory and all my colleagues in the Institute of Molecular Life Sciences. My special thanks go to Lukas Weber, Malgorzata Nowicka, Stephan Schmeing and Katharina Hembach for many constructive discussions. I am grateful to Helen Lindsay, Charlotte Soneson and former members Charity W. Law and Andrea Riebler who gave me productive suggestions and useful comments.

Finally, I would like to thank everyone who helped and supported me during my PhD studies.

Zurich, March 2017

Xiaobei Zhou

Zusammenfassung

In den letzten 4-5 Jahren sind die statistischen Methoden für die differentielle Expressionsanalyse von Hochdurchsatz-Sequenzierungsdaten (z.B. RNA-Sequenzierungsdaten) stark ausgereift und verfeinert worden. Die Zielstellung meiner 4-jährigen Doktorarbeit war es, angepasste statistische Methoden zu verwenden um robuste Analyse-Lösungen zur Interpretation von (hochdimensionellen) Hochdurchsatz-Sequenzierungsdaten zu entwickeln und existierende Ansätze zu verbessern.

Zunächst werde ich die Doktorarbeit und den nötigen wissenschaftlichen Hintergrund vorstellen. Dies umfasst unter anderem die RNA-Sequenzierungs (RNA-seq)-Technologie, die gängigen Methoden für die RNA-seq Datenanalyse, den konzeptuellen Rahmen zur Modellierung von Zähldaten (z.B. ***edgeR***), Ausreisser und robuste Methoden sowie die aktuellen Entwicklungen und Fortschritte in der differentiellen Expressionsanalyse von Transkripten. Zusätzlich werden meine Forschungsziele und die damit verknüpften Herausforderungen dargestellt. Eine kurze Zusammenfassung der einzelnen Kapitel werden im Nachfolgenden präsentiert.

Im Anschluss an die Einführung präsentiere ich eine robuste statistische Methode zur differentiellen Expressionanalyse von RNA-seq-Daten unter Verwendung von Observation-Weights. Der Effekt von Ausreissern in RNA-seq-Daten kann nicht ignoriert werden, weil ausreichende Stichprobengrössen aufgrund der hohen Kosten für RNA-seq oft nicht möglich sind. Allerdings ist (statistische) Robustheit bei kleinen Stichproben extrem schwer zu erzielen. Derzeitige differentielle Expressions (DE)-Methoden implementieren im Besonderen eine komplexe Struktur um Informationen des ganzen Datensatzes nutzen zu können und dadurch die statistischen Schlussfolgerungen zu verbessern. In dieser Dissertation wurde eine neue robuste statistische Methode im Rahmen des Generalisierten Linearen Modells (GLM)/Moderated-Dispersion-to-Trend entwickelt um die Auswirkungen von Ausreissern in RNA-seq zu dämpfen. Zusätzlich wurde ein offenes Programmiergerüst für Simulationen von Zähldaten, basierend auf R Code, entwickelt um die Reproduzierbarkeit zu fördern und eine Anpassung des Benchmark-Test zu ermöglichen wenn neue Methoden entwickelt werden. Weiter Informationen dazu gibt es in Kapitel I.

Als Nächstes werde ich ein R-Paket names ***benchmarkR*** vorstellen, das mithilfe von verschiedenen maßsgeschneiderte metrische Grafiken ein allgemeines Benchmarking von genomweiten Methoden für Datensätze mit einer unabhängigen Wahrheit (z.B. Simulationen) erlaubt. Wenn eine neue statistische Methode zur Entdeckung von differentielle Expression in Genomdatensätzen entwickelt wird, sind Vergleiche mit bestehenden Methoden, die eine Leistungsverbesserung zeigen, notwendig, aber schwierig und zeitaufwändig. Die Motivation bei der Entwicklung des benchmarkR-Pakets war den Aufwand für solche Vergleichen zu verringern und eine objektive, umfangreiche und visualisierte Lösung für das Benchmarking von genomweiten Methoden zur Verfügung zu stellen. Die grösste Neuerung des Pakets ist eine flexible grafische Darstellung der True Positive Rate (TPR) versus erzielter False Discovery Rate (FDR), in der die zwei wertvollsten Metriken, TPR und FDR, gleichzeitig gezeigt werden. Insbesondere gibt diese Darstellung Auskunft darüber wie gut die Methoden kalibriert sind,

das heisst, ob die evaluierten Methoden die geschätzte FDR (z.B. 5% FDR) erreichen. Genauere Informationen sind in Kapitel II beschrieben.

Anschliessend habe ich die Leistungsfähigkeit von differentiellen Expressionsmethoden für Zähldaten untersucht wenn Gene in nur einer der Versuchsbedingungen exprimiert sind. Fast alle RNA-seq Experimente enthalten eine Teilmenge von Genen, die keine nachweisbaren Reads in einer der getesteten Versuchsbedingungen haben aufgrund von sehr geringer oder fehlender Expression, dennoch sollten diese Gene auffindbar sein. Diese Teilmengen werden mit "Zero-Count" bezeichnet. Ein Bericht von Rapaport und seinen Mitautoren in Genome Biology, in dem behauptet wird, dass Zähldaten-basierte Methoden (z.B. ***edgeR***) bei Zero-Count Daten eine geringe Leistungsfähigkeit zum Entdecken von differentiellen Expressions Merkmalen haben, hat unsere Aufmerksamkeit auf sich gezogen. Wir haben die Auswirkungen von Zero-Count auf echte Datensätze und simulierte Zähldaten-basierte Datensätze analysiert und die Ergebnisse in einem Correspondence Artikel zusammengefasst (Kapitel III).

Zum Schluss präsentiere ich ein frühes Projekt in dem Normalisierungsmethoden für microRNA-Sequenzierungsdaten evaluiert wurden. Funktionieren die aktuell gängigen Normalisierungsmethoden für messenger RNA Sequenzierungs (mRNA-Seq)-Daten, wie zum Beispiel Trimmed Mean of M-values ("TMM"), auch für micoRNA Sequenzierungsdaten? Um diese Frage zu beantworten, haben Garmire und Subramaniam in ihrem Papier einen Vergleich zwischen verschiedenen Normalisierungsmethoden für microRNA Sequenzierungsdaten gemacht (einschließslich spezialierter Methoden für mRNA-seq Daten und anderer populärer Methoden für biologische Daten) und festgestellt, dass die Leistung von TMM schlecht ist. Wir stimmen diesem Ergebnis nicht zu und präsentieren unsere Neuanalyse in einem Letter to the Editor (Kapitel IV).

Abschliessend diskutiere ich einige offene Problemstellungen in Bezug auf die robuste Methode, die wir entwickelt haben, berichte über meine Fortschritte in der Forschung und meinen Beitrag zur DTE-Analyse und gebe einen Ausblick (Kapitel V).

Abstract

Statistical methods for differential expression analysis of high-throughput sequencing data (e.g., RNA-sequencing data) have become quite mature and refined during the last 4-5 years. The overarching goal during my 4 years' PhD research was to develop robust analysis solutions and improve existing approaches for interpreting (high-dimensional) high-throughput sequencing data using tailored statistical methods.

First, I introduce the dissertation. Some necessary scientific background are introduced, including RNA-sequencing (RNA-seq) technologies, popular methods for RNA-seq data analysis, the main conceptual frameworks for modeling of count data (e.g., ***edgeR***), outliers and robust methods and current development and progress in differential *transcript* expression analysis. Additionally, my research objectives and challenges in my research field are presented. A short summary of following chapters are subsequently represented.

Following the **Introduction**, I present a robust statistical method for detecting differential expression in RNA-seq data using observation weights. The effect of outliers in RNA-seq data cannot be ignored because sufficient sample sizes are often not possible due to the high cost of RNA-seq. However, (statistical) robustness is extremely challenging in small samples. In particular, current differential expression (DE) methods implement a complex structure for sharing information across the whole dataset to improve inferences. A new robust statistical method was designed to dampen the effect of outliers in RNA-seq within the existing generalized linear model (GLM)/moderated-dispersion-to-trend framework. Additionally, an open framework of count-based simulation based on R code was developed to promote reproducibility and allow the benchmark to move forward as new methods are developed. Further information can be found in Chapter I.

Next, I introduce an R package named ***benchmarkR*** for general benchmarking of genome-scale methods for datasets that have an independent truth (e.g., simulations), via several customized metric plots. When a new statistical method for detecting differential expression in genomic datasets is developed, comparisons against existing methods that show improved performance are necessary but difficult and time-consuming. The motivation for developing the ***benchmarkR*** package was to reduce the burden during comparisons and provide an objective, comprehensive and visualized solution for benchmarking genome-scale methods. The main innovation of the package is a flexible true positive rate (TPR)-versus-achieved false discovery rate (FDR) plot, representing the two most valuable metrics TPR and FDR at the same time. Notably, this plot can be used to show information on how well the methods are calibrated. That is, whether the evaluated methods achieve the estimated FDR (e.g., 5% FDR). More detailed information is described in **Chapter II**.

Subsequently, I made some contribution to investigating the performance of count-based differential expression methods when genes are expressed in only one condition. Almost all RNA-seq experiments include a subset of genes that have no detectable read counts in one of the tested conditions due to very low or lack of expression, but they appears to be detected. These subsets are so-called "zero-counts". A report in Genome Biology from Rapaport and

co-authors that claimed a low performance of count-based methods (e.g., ***edgeR***) to detect differential expression features for zero-count data drew our attention. We analyzed the effect of zero-count on real datasets and simulated count-based datasets and summarized the results as a Correspondence article (**Chapter III**).

Then, I present an early project evaluating normalization methods for microRNA sequencing data. Do currently popular normalization methods used for messenger RNA sequencing (mRNA-seq) data, such as Trimmed Mean of M-values ("TMM"), still work well for microRNA sequencing data? To address this question, Garmire and Subramaniam in their paper made a comparison between several normalization methods (including methods specialized for mRNA-seq data and other popular methods for biological data) for microRNA sequencing data and concluded a poor performance of TMM. We disagreed with their results and presented our reanalysis as a Letter to the Editor (**Chapter IV**).

Finally, I discuss some issues related to the robust method we developed and report my current research progress and contribution to DTE analysis as further perspectives (**Chapter V**).

Thesis outline

Introduction

Chapter I: **Robustly detecting differential expression in RNA sequencing data using observation weights**
Xiaobei Zhou, Helen Lindsay and Mark D. Robinson
Paper published in *Nucleic Acids Research (2014), 42, pp. e91*

Chapter II: **benchmarkR: an R package for benchmarking genome-scale methods**
Xiaobei Zhou, Charity W. Law and Mark D. Robinson
Paper published in *bioRxiv (2015)*

Chapter III: **Do count-based differential expression methods perform poorly when genes are expressed in only one condition?**
Xiaobei Zhou and Mark D. Robinson
Paper published in *Genome Biology (2015), 16, 222*

Chapter IV: **miRNA-Seq normalization comparisons need improvement**
Xiaobei Zhou, Alicia Oshlack and Mark D. Robinson
Paper published in *RNA (2013) 19 (6), 733-734*

Chapter V: **Discussion and perspectives**

Contents

Introduction **14**
1. Background . . . 14
1.1. RNA-seq . . . 14
1.2. Current popular methods for RNA-seq analysis . . . 17
1.3. The count model framework of ***edgeR*** . . . 22
1.4. Outliers in RNA-seq data and robust methods . . . 25
1.5. Zero-counts in RNA-seq data . . . 26
1.6. DTE analysis . . . 26
2. Research objectives and challenges . . . 29
2.1. Research objectives . . . 29
2.2. Objective 1: a robust statistical method for detecting differential expression in RNA-seq data . . . 29
2.3. Objective 2: an open framework based on R code for benchmarking genome-scale methods . . . 30
2.4. Objective 3: zero-counts . . . 30
2.5. Objective 4: DTE analysis . . . 30
2.6. Research challenges . . . 31
3. Thesis Summary . . . 31

I. Chapter I **39**
Robustly detecting differential expression in RNA sequencing data using observation weights . . . 39

II. Chapter II **51**
benchmarkR: an R package for benchmarking genome-scale methods . . . 51

III. Chapter III **57**
Do count-based differential expression methods perform poorly when genes are expressed in only one condition? . . . 57

IV. Chapter IV **63**
miRNA-Seq normalization comparisons need improvement . . . 63

V. Chapter V **67**
Discussion and perspectives 68
1. Discussion 68
1.1. Robust M-Estimator or weighted likelihood estimator for robust method? 68
1.2. Could poor FDR control be linked to the inappropriate observation weights of our robust method? 68
2. Perspectives 68
2.1. My contribution to DTE analysis 68

Introduction

1 Background

1.1 RNA-seq

1.1.1 A short summary of RNA-seq technology

Ribonucleic acid (RNA) plays an important role in carrying the genetic information of organisms (along with deoxyribonucleic acid (DNA)). RNA usually appears as a single-stranded chain of nucleotides; each nucleotide is made up of any one of four nitrogenous bases, including adenine (A), cytosine (C), guanine (G) and uracil (U). RNA plays a central role as a bridge connecting between DNA and proteins, famously known as the central dogma of molecular biology. The central dogma firstly introduced by Francis Crick [9] explains the flow of genetic information in biological systems from DNA to RNA to protein. This flow can be summarized as two steps: transcription and translation. During the transcription process double-stranded DNA is converted to single-stranded RNA; RNA carries information from DNA (i.e., messenger RNA (mRNA) is produced.). During translation, mRNAs find a way to the ribosome, where they are translated into protein.

RNA-sequencing (RNA-seq) technology has revolutionized the exploration of the presence and quantification of RNA in biological samples. Generally speaking, current mature methods of RNA-seq that can produce millions of reads (i.e., fragments) share the following critical steps: i) RNA extraction from biological samples, ii) selection of subpopulation of interest (e.g., polyA-enriched or ribosomal RNA depletion), iii) reverse transcription into complementary DNA (cDNA) and iv) library preparation and fragmentation. Notably, these reads can be basically partitioned as two types: i) single-end reads that sequence only one end of a fragment and ii) paired-end reads that sequence both ends. RNA-seq is a reverse-engineering technology to quantify the features of biological interest in biological samples from the digital readout (i.e., counts of fragments) for both well-known and less characterized organisms. For well-annotated organisms, the observed fragments (reads) are put in the context of existing annotation (i.e., the reads are aligned/mapped to a reference genome.). In the absence of annotation, catalogs of transcripts are generated by (*de novo*) assembly [16]. Compared to the earlier biological technologies, such as microarrays, RNA-seq offers "an open system, higher resolution, lower relative cost and less bias" [55]. Using RNA-seq technology, it became practical and affordable to study entire transcriptomes—the complete set of RNA transcripts expressed by one cell or a population of cells. RNA-seq is commonly used for detecting differential expression in comparative experiments for delineating alternative splicing patterns, RNA editing, discovering novel transcripts and profiling of allele-specific expression.

This thesis focuses on the most common application of RNA-seq: detecting changes in expression between experimental conditions or treatments. In this case, the primary goal of RNA-seq experiments is to find which biological features (e.g., genes) are statistically significant (i.e., the differences are larger relative to their random variations). To satisfy this task, it is necessary to carefully consider the experimental design. Two basic properties cannot be ignored: sufficient replication and sequencing depth. However, considering that RNA-seq remains expensive, researchers have to consider the tradeoff between getting more reads (depth) or more sample replicates within a limited budget. Not surprisingly, recent research has suggested that increasing the number of replicate samples is preferred to increased sequencing depth for improving detection power [38]. Additionally, tailored statistical methods for detection of differential expression for RNA-seq data should be applied, which can handle the biolog-

ical variability in count data, and allow a much broader class of experimental designs to be analyzed, such as batch effects (discussed in Section 1.2.3 and 1.3).

1.1.2 RNA-seq analysis

A typical RNA-seq data analysis workflow (presented in Figure 1) can be simplified as four steps: quality control (QC), mapping, quantification and differential analysis. In the QC step, the raw data is evaluated by a series of metrics for quality assessment, including sequence quality, sequencing depth, guanine-cytosine (GC) bias and so on. In the mapping step, the raw reads (cDNA fragments) produced by the sequencing machine are mapped to a reference genome or (assembled) transcriptome by an alignment algorithm, such as ***TopHat*** [51] or ***bowtie*** [23]. In the quantification step, there are two distinct approaches, either: exon-union counting where the aligned reads that overlap the target genomic regions-exon-union sets (e.g., gene) are counted; or transcript abundance estimation that estimates the expected count of the aligned reads originating from a transcript by a probabilistic manner (e.g., multinomial model). Notably, for estimating transcript abundance, some tools can skip the traditional mapping step to quantify abundances (see Section). In the step of differential analysis, the (expected) count-table constructed previously can be analyzed in different ways, depending upon the biological questions of interest.

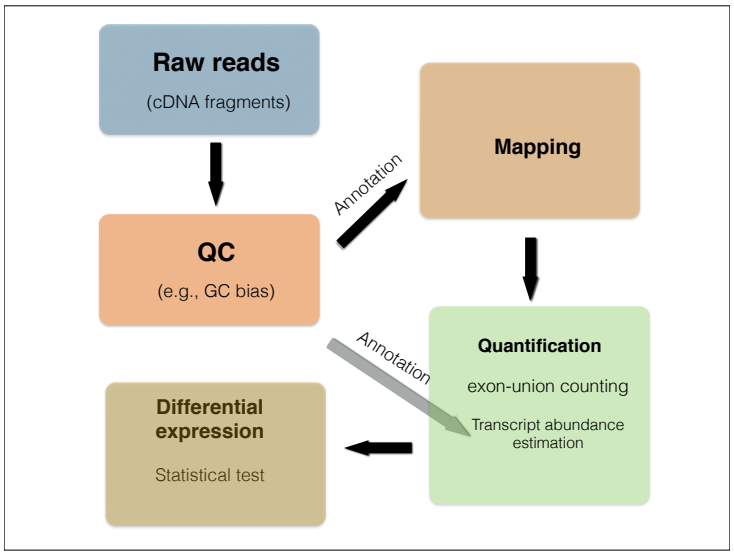

Figure 1.: RNA-seq data analysis workflow.

RNA-seq analyses can be used to address diverse biological questions, such as allele-specific expression, RNA editing, gene co-expression networks, rare transcript detection but most commonly, for differential expression and differential splicing between experimental conditions or treatments. Focusing on comparing the transcriptional output between different conditions, we can partition most of the RNA-seq studies into either: differential gene/transcript expression (DGE/DTE) or differential transcript/exon usage (DTU/DEU) studies. Their relationship (illustrated in Figure 2) will be discussed in the following sections.

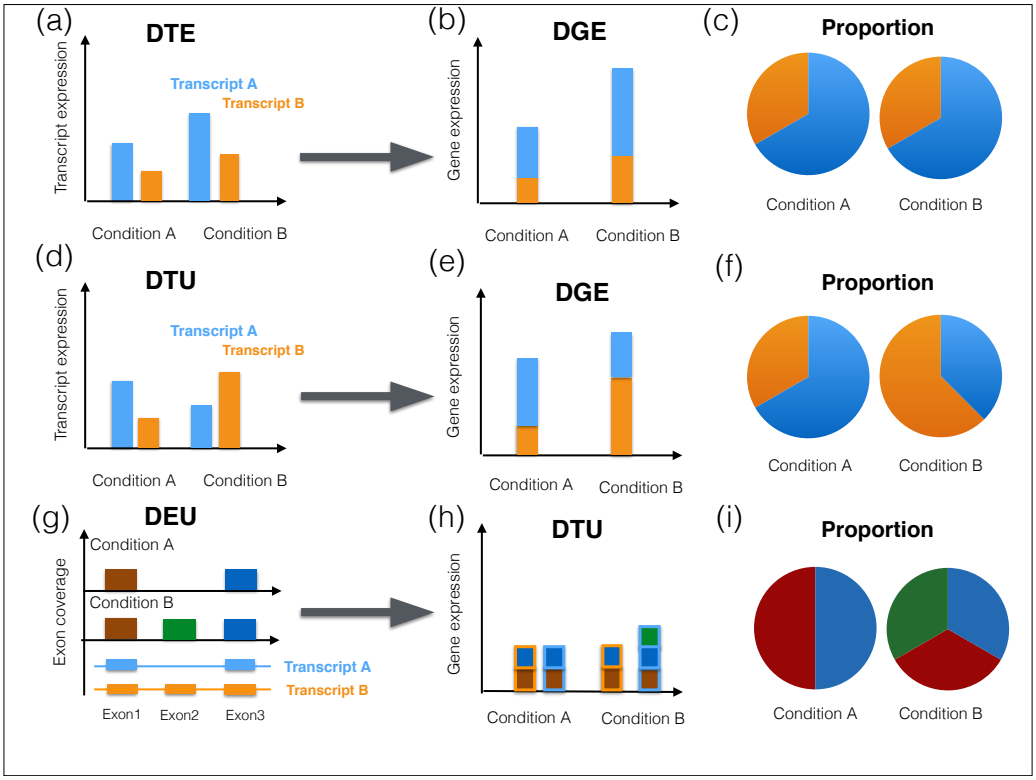

Figure 2.: Schematic illustration of relationship among DGE, DTE, DTU and DEU between conditions A and B, for a gene with two transcripts (isoforms). (a), (d) and (g) schematically illustrate cases of DTE, DTU and DEU; for the cases of DTE and DTU, their integrated gene level expression are correspondingly shown in (b) and (e); for the case of DEU, the illustration in form of DTU is presented in (h); and the proportions of cases of DTE, DTU and DEU are shown in (c), (f) and (i).

Differential gene/transcript expression

A feature (e.g., gene or transcript) is declared differentially expressed if an observed change in (normalized) expression across different experimental conditions is unlikely to occur by chance under some specification of a null model. Models for count data are usually preferred for differential expression (DE) analysis because the data type of final output from an RNA-seq experiment is count data, although there are variously alternative solutions (e.g., the approach that transforms the count data into the ordinary linear model [24]). As shown in Figure 2 (a) and (b), DTE analysis is focused on whether individual transcripts have changed across experimental conditions but requires estimating transcript abundance. Thus, there are additional challenges since there are the uncertainties introduced during estimation (discussed in Section 1.6.2). By contrast, DGE analysis looks at whether the overall transcriptional output has changed to find out which genes are up- or down-regulated in each condition. The overall transcriptional output (i.e., gene abundance) can be achieved by either aggregating all the transcripts per million (TPMs) within a gene or directly counting aligned reads at the gene level. The approach of simple summing all the reads overlapping the target gene (i.e., union counting) may not accurately reflect the true gene expression in case of multiple isoforms within a gene, since for the same level of expression, longer transcripts contribute more reads on average [50]. DTE may or may not imply DGE: it depends on all the information of transcripts within a gene. For example, as shown in Figure 2 (d) and (e), when two transcripts are both differentially expressed but inversely regulated, the difference in the gene level may not be large enough to imply DGE. On the other hand, DGE must imply one or more tran-

scriptional changes.

Differential transcript/exon usage

A modified version of the definition of relative usage of a feature (e.g., transcript or exon) can be found from the vignette of ***DEXSeq*** [40]:

$$\frac{\text{expression of a feature}}{\text{expression of the gene containing this feature}}. \quad (1)$$

DTU/DEU analysis considers changes in the relative usage of transcripts/exons across different experimental conditions. This is shown in Figure 2 (d), (f), (g) and (i). DTU and DEU both emphasize that feature proportions change across different experimental conditions, where features can be: transcripts or exons (or non-overlapping exon bins [4]). Analysis of DTU focuses on the proportions of the individual transcripts in comparative experimental conditions within a gene, while DTE analysis only looks at an observed change of individual transcripts across experimental conditions regardless of whether other transcripts within the gene are changing [48]. For instance, in Figure 2 (d), we observed two transcripts that are changing inversely resulting in a proportional change in Figure 2 (f), while in Figure 2 (c), the proportion change may not be observed. Regardless of what happens at the overall gene level, DTE may not imply DTU when the proportions of the individual transcripts do not change or the proportional changes do not achieve a certain level. On the other hand, DTU implies one or more changes in transcript expression [1, 48]. In Figure 2 (d), the expression of transcript A and B change accompanied by DTU. For relationship of DTU and DEU, DEU is an "indicator" of DTU: DEU is sufficient for DTU. For instance, in Figure 2 (d), the exon 2 will imply the change of proportions of transcript A and B resulting in DTU. For relationship between DTU/DEU and DGE, DTU/DEU can result in DGE if the sum of overall transcript outputs differs large enough across experimental conditions. In Figure 2 (e), the overall change at the gene level is small, since its corresponding transcripts are changing inversely. On the other hand, DTU/DEU may still exist, regardless of presence/absence of DGE.

1.2 Current popular methods for RNA-seq analysis

There are many bioinformatics tools available for processing and analyzing RNA-seq data and the number is still increasing rapidly. Selected representative tools are introduced as follows.

1.2.1 Alignment algorithms

Once sequencing reads are generated, the next task is to align raw reads to a reference genome or (assembled) transcriptome. This task needs to satisfy the following basic requirements: i) it is robust allowing for a certain level of insertions/deletions caused by genomic variations, mismatches caused by sequencing errors during calling the sequencer and other errors from the experimental preparation (e.g., RNA priming preference [18]). ii) it can identify splice junction structures (or non-continuous genomic regions), which is of particular interest for studying alternative splicing. Combining those requirements together makes the procedure of aligning reads very computationally intensive. Two representative and popular used align-

ment algorithms are represented here.

TopHat

TopHat is a widely used sequence alignment algorithm for aligning sequencing reads to a reference genome or (assembled) transcriptome [51]. ***TopHat*** was developed based on a well known short DNA sequence aligner ***bowtie*** [23] to provide an ability to align RNA-seq reads without relying on known spliced junctions. The basic idea is summarized as the following steps: i) aligning non-junction reads to a whole genome using the aligner ***bowtie*** and assembling the mapped reads by the algorithm, ***Map*** [27], to generate consensus sequence regions; ii) finding all potential flanking donor/acceptor junction sites within neighboring regions (i.e., exons) to identify the splice junction structures; and, finally, mapping initially unmapped reads from ***bowtie*** to these splice junctions. The current version of ***TopHat***, named ***TopHat2***, improves the performance in several cases, including various lengths of reads, insertions, deletions and gene fusions [22].

STAR

STAR was developed to solve the computational bottleneck of the task to align millions of reads [10]. Similar to ***TopHat***, ***STAR*** can identify splice junction structures. However, the principle of ***STAR*** for alignments is completely different from ***TopHat***. ***STAR*** can directly align non-contiguous reads to splice junction structures, while ***TopHat*** is an extension of ***bowtie*** relying on aligning non-junction reads to generate consensus sequence regions. The algorithm of ***STAR*** can be summarized as 2 steps: seed searching and stitching steps. For seed searching step, ***STAR*** will find the seed (i.e., the fragment of the read) mapped to the donor junction site by searching a Maximal Mappable Prefix (MMP) [10], then the MMP search is repeated for the unmapped positions of the read to the acceptor junction site. For the stitching step, ***STAR*** will build the entire alignments by stitching all the seeds that are generated by the first step together. This approach has been shown to give a great improvement in computational speed [12]. Additionally, ***STAR*** can provide estimates of the relative probabilities of the alignments, in which a read can be mapped to multiple loci. This alignment algorithm became popular and widely used since the ENCODE project was started [5].

1.2.2 Tools for RNA-seq quantification

There are a range of approaches for RNA-seq quantification, including i) simple counting of the aligned reads overlapping the target genomic regions (e.g., ***HTSeq*** [2] and ***featureCounts***) and ii) estimating or quantifying the abundances of individual transcripts through probabilistic models that probabilistically assign the alignments to isoforms of a gene (e.g., ***Cufflinks*** [52], ***RSEM*** [26], ***kallisto*** [6] and ***Salmon*** [33]). The simple counting approach and its downstream analysis pipeline are mature but this approach may not reflect the true expression change [50]. Compared to the simple counting approach, estimating transcript abundance provides a higher resolution at the transcript level, but it requires paying a special effort to process ambiguously mapping reads (see Section 1.6.2). Moreover, this approach may not be practical for a complex situation where a gene has many isoforms. One extreme example is

the DSCAM gene in Drosophila melanogaster, which has 38,016 isoforms [32]; no algorithm can provide enough statistical power to estimate abundance for all the possible isoforms. Selected programs for RNA-seq quantification are presented as follows:

HTSeq (htseq-count) and *featureCounts*

htseq-count (implemented in ***HTSeq***) and ***featureCounts*** are both widely used to count the overlap of reads with a catalog of features (i.e., annotation relative to reference genome). A read is considered to *overlap* a feature if any overlap (i.e., hit) between the read and the feature is found [29]. ***HTSeq*** and ***featureCounts*** count the number of reads from biological samples overlapping the target features of interest given SAM/BAM files containing aligned reads and a GTF/GFF file recording detailed information of biological features (e.g, transcripts or genes). They can provide the result in the form of a count-table that records the expression of each feature for each biological sample. This count-table can be provided to downstream analysis (e.g., DGE analyses using methods such as ***DESeq2*** [30] or ***edgeR***). For reads overlapping multiple features, both tools recommend excluding them by default during counting.

Cufflinks2 and *RSEM*

Cufflinks2 is used to assemble transcripts and estimate transcript abundance for RNA-Seq data [53]. For sequencing data where the genome sequence is unknown, the algorithm can assemble the aligned reads into a parsimonious set of transcripts (i.e., minimal set of transcripts to explore the dataset) without relying on prior knowledge of genomic annotation. For data with a known genome sequence in a situation where annotation suffers from low quality or is incomplete, ***Cufflinks2*** can assemble novel transcripts in the context of an existing annotation to improve annotation quality [41]. For estimating transcript abundance, ***Cufflinks2*** firstly fits alignments to each RNA molecule, which is based on the assumption that each alignment follows a normal distribution [53]. Then ***Cufflinks2*** probabilistically assigns alignments to gene isoforms given the probability predicted by the RNA molecule. Finally ***Cufflinks2*** estimates (relative) transcript abundance by maximizing the likelihood of all possible sets of assignments. The variability of assigning alignments can be considered as the uncertainty of transcript abundance estimation.

For estimating transcript abundance, ***RSEM*** uses a similar strategy to assign alignments to isoforms, but uses an advanced estimator, which is the expectation-maximization (EM) algorithm instead of maximum likelihood to estimate abundances [26]. This approach is computationally intensive but has a better estimation accuracy in the case where reads are ambiguously mapping to multiple isoforms due to high similarities among transcripts [26].

BitSeq

BitSeq uses a Bayesian approach for estimation of transcript abundance. It is based on a probabilistic model frame that probabilistically assigns alignments to isoforms, similar to the model of ***Cufflinks2*** and ***RSEM***, to account for ambiguous alignment caused by complex isoform structures. The posterior distributions of model parameters are generated by a Markov

chain Monte Carlo (MCMC) algorithm. Compared to ***Cufflinks2*** and ***RSEM***, ***BitSeq*** is computationally intensive.

Sailfish, *Salmon* and *kallisto*

Some quantification methods, which are so-called alignment-free methods (e.g., ***Sailfish*** [34], ***Salmon*** [33] and ***kallisto*** [6]), are designed to estimate transcript abundance without relying on alignments provided by an external aligner. In fact, these quantification methods still need to make a similar calculation as alignment to determine the concordance between reads and transcripts, but they do not do the full alignment that requires spending the computational resources for determining the optimal alignment, and hence the speed of these methods is fast. The idea of these methods is to transform the original data (including raw reads and reference transcripts) to small units in terms of k-mers and get a summary statistic based on k-mers. Here, k-mers refers to all the possible subsequences of a read or transcript of length k. ***Sailfish*** firstly builds a k-mer based index of reference transcripts in the form of a hash table, then quantifies abundances by matching k-mer unit reads using the hash table. This approach can dramatically reduce the cost of computational resources compared to the full alignment. However, it has been shown to lead to a loss of estimation accuracy in the case of a simulated dataset [6] and in the case of complex mixtures of isoforms [33]. To address this practical issue, ***Salmon***, the successor of ***Sailfish***, and ***kallisto*** were developed to reduce errors while still keeping a fast computational speed. Both programs focus on reducing the errors caused by the usage of k-mer based approach: ***Salmon*** uses lightweight alignments based on chains of maximal exact matches instead of k-mer hashing and adds two-phase algorithms (online and offline inference) to improve the accuracy [33]. ***kallisto*** applies pseudoalignments, which computes the k-mer units of reads to paths in whole transcripts (constructing a transcriptome De Bruijn graph (T-DBG)), to remove the redundancy of k-mers from computation for reducing the errors [6].

1.2.3 Tools for differential expression (DE) analysis

Statistical methods for DE analysis of RNA-seq experiments across different experimental conditions or treatments have become mature after several years of development. There are various methods available for the biological and statistical communities, including parametric and non-parametric methods. Compared to non-parametric methods (e.g., ***NOISeq*** [49] and ***SAMseq*** [28]), parametric methods (e.g., ***edgeR*** [42] and ***DESeq2*** [30]) are more popular, since parametric models following an appropriate model assumption are more powerful for most cases of RNA-seq data in which replicates of experiments are limited. For parametric models, they can be mainly classified as either: i) methods directly working on the count data, which are usually based on count regression models, such as the negative binomial (NB) model (e.g., ***edgeR***) and ii) methods transforming the counts and propagating transformed values into the ordinary linear regression model (e.g., ***voom*** [24]). Focusing on statistical inference, the methods can be partitioned into i) frequentistic inference methods (e.g., ***edgeR***) using hypothesis tests (e.g., likelihood ratio test) to detect DE features, ii) Bayesian inference methods (e.g., ***baySeq*** [20] and ***BitSeq***) that can directly provide false discovery rates to identify DE features.

Developing robust and tailored statistical methods to detect DE features is one of the major research directions of the Robinson Statistical Bioinformatics group. In particular, my supervisor, Prof. Mark Robinson co-authored one of the most widely used R-packages, ***edgeR***.

Improving statistical methods for DE analysis is the core task of my PhD study and the core content of this dissertation. In the following Sections 1.2.3, 1.3.1, 1.3.2 and 1.3.3, we will introduce selected popular DE methods, the NB model framework and advanced methodologies of ***edgeR*** in detail.

edgeR

edgeR is a widely-used package for RNA-seq differential expression (with biological replication) within the ***R/Bioconductor*** software development project [14]. It implements a range of statistical methodologies, including the NB model in generalized linear model framework, exact tests, likelihood ratio tests and quasi-likelihood tests. Additionally, several specific advanced approaches, such as adjusted profile likelihood and moderated NB dispersion estimation, are developed to handle the challenges caused by genome-scale count data. As well as RNA-seq, this tool can also be applied to any other genome-scale count data, such as ChIP-seq, where the aim is to find differentially regions of the genome.

The tool is designed to calculate statistical evidence for changes in expression levels across tens of thousands of features in the case where the degrees of freedom of the regression model of each feature are limited (e.g., $\not> 8$). Since increasing the sample size (i.e, biological replicates) is not usually possible due to the relatively high cost of RNA-seq, an alternative solution, learned from experience in analyzing microarray data, is to use an empirical Bayesian approach that shares information over the whole dataset to improve inference. In particular, ***edgeR*** applies an advanced approach that allows sharing of information (in terms of dispersion estimation) between "neighbours" of features that have similarity in average expression. This tool moderates dispersion estimates towards a trended-by-mean estimate by maximizing adjusted profile likelihood (APL) [31] (see Section 1.3.2 and 1.3.3). ***edgeR*** makes parametric assumptions for inference; in particular, the NB distribution is assumed in the context of a generalized linear model [31], thus allowing complex experimental designs to be analyzed.

DESeq2

DESeq2 [30], as the successor of ***DESeq*** [3], is similar in many respects to ***edgeR***. For dispersion estimation, ***DESeq2*** uses a similar approach to ***edgeR***, shrinking the individual dispersion estimation toward the fitted curve that is estimated from the individual dispersion against the expression value. ***DESeq2*** is also based on the NB model via a GLM framework. However, there are several distinctions between them: i) ***DESeq2*** takes additional shrinkage for fold-change estimation to improve stability of low read counts and thus applies Wald tests instead of likelihood ratio tests for tests of significance [30]. In addition, ii) ***DESeq2*** and ***edgeR*** use different normalization methods for RNA-seq data: Trimmed Mean of M-values ("TMM") for ***edgeR*** and median ratio method introduced by Anders and Huber [3] for ***DESeq2***.

Cuffdiff2

Cuffdiff2 [50], a program implemented in ***Cufflinks2***, can be used for DE analysis. For DTE analysis, its model based on a beta NB distribution that can be interpreted as a mixture of NB distributions can account not only for biological and technical variability, but also for the uncertainty of transcript estimation. The uncertainty is estimated by ***Cufflinks2***. If there is

no uncertainty, it reduces to a NB model. For DGE analysis, it should follow a NB model (the details are not clear). The mean and variance parameters in a gene are estimated from a group of transcripts of this gene: the total mean is calculated by summing the corresponding (normalized) transcript abundances and the variance is obtained by summing covariances of (normalized) transcript abundances. Then the estimated mean and variance at the gene level are used for fitting the NB model.

limma-voom

voom [24], an extension of ***limma***, specializes for DE analysis of RNA-seq data. ***limma*** [45] is a powerful ***R/Bioconductor*** package for analyzing microarray data using an empirical Bayesian approach based on linear models. ***voom*** can be considered as a transformation method that puts a $log2$ transformation of the normalized counts and observation weights into the existing ***limma*** analysis pipeline. The weights for each observation estimated from the dataset are used to account for the heteroscedastic variances in transforming count data with a logarithm. Compared to methods based on the GLM framework, such as ***edgeR***, the advantage of ***voom*** is its fast speed and good false discovery control. The disadvantage is that it suffers from somewhat lower power to detect DE features (e.g., genes) in the case of small numbers of biological replicates [46].

NOISeq(BIO)

NOISeq is a non-parametric method for DE analysis for RNA-seq data with technical replicates or no replications [49]. It focuses on log fold change (i.e., log-ratio of average expression values) for the two experimental conditions and the absolute expression difference as statistics to test for significance. ***NOISeqBIO*** joined the existing model framework of ***NOISeq*** with an empirical Bayesian approach introduced by Efron et al. [11]. This approach was originally developed for microarray data in which a Z statistic was defined to measure differential expression across two experimental conditions. The distribution of the Z statistic consists of a mixture of components between the null set (when a feature is not differentially expressed between two conditions) and the DE set (when a feature is differentially expressed) with their corresponding probabilities (i.e., weights). Then, the local false discovery rate (FDR) based on the mixture distribution and distribution of the null set is calculated. The innovation within ***NOISeqBIO*** is to define the Z statistic for RNA-seq data as an equal combination between log fold change and the expression difference with an adjustment for biological variance to reflect the variations in biological replicates.

1.3 The count model framework of edgeR

1.3.1 The NB model in generalized linear models (GLM) framework

The GLM formula for fitting count data Y_{ij}, representing the read counts in sample j for feature i, with a canonical logarithm link is as follows:

$$log(\mu_{ij}) = X\beta_i + \log N_j, \tag{2}$$

where X is the design matrix containing the covariates (e.g., experimental conditions, batch effects, etc.), β_i is a vector of regression parameters and N_j is the (effective) library size for sample j. In our setting, Y_{ij} is assumed to follow a NB distribution with mean μ_{ij} and dispersion ϕ_i, denoted by $Y_{ij} \sim NB(\mu_{ij}, \phi_i)$.
This framework within ***edgeR*** (and ***DESeq***) has been constructed as a "two-stage estimation": firstly estimating the dispersion ϕ_i; then estimating β_i [31]. Given an estimated dispersion $\hat{\phi}_i$ that is assumed known up to a constant (discussed in Section 1.3.2 and 1.3.3), the estimated value, $\hat{\beta}_i$, can be obtained by the iteratively re-weighted least squares (IRLS) algorithm.

1.3.2 Adjusted profile likelihood

For this model, a key question is how to estimate dispersion reliably in presence of a nuisance parameter (regression parameter β). An early strategy using conditional likelihood has been shown to give an excellent performance but cannot be implemented within a GLM framework [43, 44]. Instead, approximate conditional inference introduced by Cox and Reid [8] is used here to remove the effect of the nuisance parameter. The adjusted profile likelihood (APL) for the dispersion ϕ_i, penalized for the estimation of the regression parameters, β_i, is presented as follows:

$$APL_i(\phi_i) = \ell(\phi_i; \mathbf{y}_i, \hat{\beta}_i) - \frac{1}{2} \log |\mathcal{I}_i|, \tag{3}$$

where $\mathbf{y}_i$ is the vector of counts for feature i, $\hat{\beta}_i$ is the estimated coefficient vector, $\ell()$ is the log-likelihood function, $\mathcal{I}_i$ is the Fisher information matrix and $|.|$ is the determinant.

1.3.3 Moderation: APL via weights

Due to the limited sample size of typical RNA-seq experiments, some further efforts to improve estimate (e.g., dispersion) reliability are necessary. An empirical Bayesian strategy seems a good solution for RNA-seq count data and has been successful for microarray data [45]. However, empirical Bayesian approach cannot easily be implemented in the NB model, since there is no conjugate prior distribution for estimating the NB dispersion [7]. An alternative solution is to moderate estimates toward a common trend via weighted likelihood. The weight in the weighted likelihood is equivalent to a prior in the empirical Bayesian distribution [7]. In the ***edgeR*** framework, the strategy to accomplish moderation for the dispersion is by squeezing the (genewise) individual dispersion toward a trended dispersion that is a smooth curve between individual dispersion and abundance. This approach involves maximizing the linear weighting of the individual likelihood and the trended likelihood, the two terms, respectively, in:

$$\arg\max \left\{ APL_i(\phi_i) + \alpha_i \cdot APL_{trend}(\phi_i) \right\}, \tag{4}$$

where α_i is a suitably chosen prior weight, $APL_{trend}(\phi_i)$ is the APL of a trend dispersion that depends on the overall level of expression. A dispersion-mean relationship commonly exists in the read counts of RNA-seq; many examples with further details can be found in the case studies of ***edgeR*** user guide. The moderated/trended dispersion-mean relationship for one example dataset is presented in Figure 3. The strategy to moderate dispersion via weighted likelihood can efficiently make use of the dispersion-mean relationship allowing a certain level of sharing of information between features with similar average expression to improve accuracy of dispersion estimation.

Additionally, the current version of ***edgeR*** can automatically estimate the degrees of freedom of prior weight α_i [7]. Simply speaking, the idea can be summarized as: firstly estimating the mean residual deviance of the GLM framework fitted to the counts Y_{ij} for feature i using a quasi-likelihood; then propagating the mean residual deviance into a scaled inverse χ^2 distribution to estimate the degrees of freedom of the prior weight.

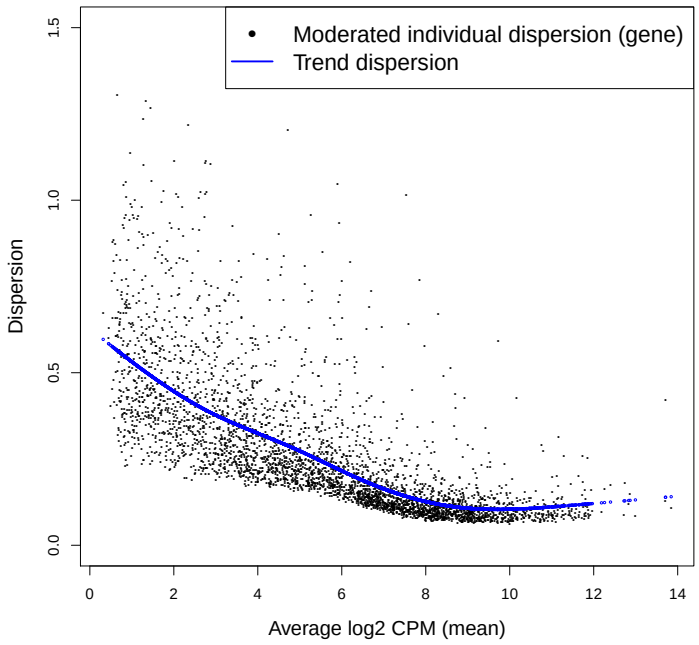

Figure 3.: The scatter plot of moderated individual (gene) and trended dispersion versus abundance (average log2 CPM) for an example of RNA-seq dataset (10 samples from the Pickrell dataset [36]). Trended dispersion is labeled in blue.

1.3.4 Strategies of sharing information in other popular methods

Beyond the moderation strategy used in ***edgeR*** (discussed in Section 1.3.3), popular strategies of sharing information to improve the accuracy of estimated dispersion/variance in other popular methods are presented as following:

Shrinkage estimation for dispersion and fold change

DESeq2 uses the same model framework of ***edgeR*** (i.e., the NB model and APL estimator). Compared with ***edgeR***, dispersion in ***DESeq2*** is additionally assumed following a lognormal distribution. The estimate of dispersion is:

$$\arg\max \{APL_i(\phi_i) + \Lambda(\phi_i)\}, \tag{5}$$

where $\Lambda(\phi_i) = \frac{-(log(\phi_i)-log(\mu))^2}{2\sigma^2}$, μ and σ are empirically estimated from the raw count data (more details can be found in [30]). Here variance σ^2 represents the strength of shrinkage that

is analogous to the prior degree of freedom in ***edgeR*** model framework (Equation 4).

Moreover, ***DESeq2*** shrinks estimated coefficients (i.e., estimated log fold changes (LFCs)) toward zero using ridge regression. The basic idea can be summarized as: estimated coefficients with high variance maybe would get a large penalty in the ridge regression. The final estimate of coefficients are achieving by minimizing the following equation:

$$L(\beta) = -(l(\beta) + 0.5\lambda||\beta||^2, \tag{6}$$

where λ, the penalty factor, is estimated from empirical quantile match procedure (the ratio between quantile of observed LFCs against theoretical quantile) [30].

Conjugate prior in classical linear models

The conjugate prior can be directly used in DE methods based on the normality assumption of the error distribution, such as ***limma***, for sharing information to improve variance estimation. The observed variance s_i^2 with degrees of freedom d_i underling true variance σ_i^2 follows a scaled χ^2 distribution:

$$s_i^2|\sigma_i^2 = \frac{\sigma_i^2}{d_i}\chi_{d_i}^2, \tag{7}$$

with a scaled-inverse χ^2 prior distribution

$$\sigma_i^2 = \frac{d_i s_0^2}{\chi_{d_0}^2}, \tag{8}$$

where s_0 and d_0 are hyperparameters. In practice, s_0 and d_0 are estimated from the marginal distribution of sample variances. And the posterior mean of σ_i given s_i can be expressed as

$$\mathbf{E}(\sigma_i^2|s_i) = \frac{d_0 s_0^2 + d_i s_i^2}{d_0 + d_i}. \tag{9}$$

This posterior value shrinking the observed variance towards the prior value with a certain level of degree of shrinkage is named as the empirical Bayes moderated variance estimator. The level of degree of shrinkage depends on the relative sizes of the observation and prior degrees of freedom. When $d_0 = \infty$, the moderated variance equals the prior value; if $d_0 = 0$, it is observed variance. The moderated variance could be given to the moderated t-statistic [45].

1.4 Outliers in RNA-seq data and robust methods

In statistics, outliers refer to the observations that are distant from other observed values [17]. In RNA-seq count data, outliers may come from various sources. In some cases, technical artifacts appear to explain some outliers (e.g. sample-specific GC content) [19]. In other cases, they seem to not follow the distribution assumption. The origin of outliers may not be traced. Maybe they come from the black box (see Section 2.6.2). Rather than discovering the origin of outliers in RNA-seq count data, we take more interest on how to treat with outliers and reduce the impact of them. Currently, there are two ways of thinking of outliers in RNA-seq data: feature-level and observation-level outliers. For feature-level outliers (e.g., gene-level outliers), Phipson et al. considered the features with extreme variances as outliers [35]. They pointed out that most of the features of RNA-seq data should follow a common prior distribu-

tion, while some of them (considered as outliers) with unusual large variances would largely affect the hyperparameter estimators. As a result, the outliers would decease the efficiency of sharing information of empirical Bayes models (i.e., reduce degrees of freedom of prior). Compared to feature-level outliers, observation-level outliers are focused at a higher resolution in which extreme observations of features are considered as potential outliers [58]. Given the types of outliers, different robust methods have been developed: i) robust hyperparameter estimation isolating the abnormal priors with extremely high variance reducing the influence of feature-level outliers [35] and ii) robust dispersion estimation via weighted likelihood using observation weights dampening the effect of observation-level outliers [58]. The second approach of robust methods is my main research topic of my PhD studies. Its details are presented in Chapter I.

1.5 Zero-counts in RNA-seq data

Zero-counts refer to the subsets of features of RNA-seq data that are expressed in only one condition or treatment. Zero-counts are a boundary condition of the parameter space (i.e., μ is 0 for one condition) and this may cause some numerical instabilities. Rapaport et al. in their comparative publication pointed out the problems caused by zero-counts [38]. In the manuscript, they found that all of the count-based DE methods based on the NB model (e.g. ***edgeR*** and ***DESeq*** [3]) exhibited the same poor performance in this case: the low correlation between signal-to-noise and confidence in differential expression as measured by adjusted P values (Figure 4). One possible explanation of this is that the generalized linear model (GLM) framework implied in ***edgeR*** and ***DESeq*** suffers from lack of robustness when many zero observations are present in one condition. GLMs require iterative fitting and more complicated dispersion estimation machinery [31]. The dispersion estimation machinery (as mentioned in Section 1.3.2) may be stable in the non-zero condition but unstable in the all-zero situation. The detail discussion related to zero-counts is shown in Chapter III. Currently, ***edgeR*** and ***DESeq2*** can handle with data contaminated with zero-counts, since log fold changes (LFCs) are always shrunk in these methods. Note that ***edgeR*** and ***DESeq2*** use different shrinkage strategies of LFCs: in ***edgeR***, a small prior counts proportion to the library sizes are added into all the raw counts to calculate LFCs; in ***DESeq2*** LFCs are estimated by a ridge regression.

1.6 DTE analysis

1.6.1 Current progress of DTE analysis

Much of the focus of the statistical methods so far has been on gene level differential expression. However, the biological and bioinformatics community is interested in looking beyond DE analyses that focus on differences only at a gene level. Current analyses of the transcriptome studies are instead increasingly focused on the connection between biological variation and the diversity and relative abundance of transcripts in a gene. In particular, researchers are taking more interest in understanding alternative splicing events and other forms of alternative isoform expression that play a significant role in tissue specific differentiation or can cause some human diseases. For instance, the splicing factor Tra2-beta1 in the human CD44 gene has been shown to be associated with breast cancer [56]. Hence, there is a great need to detect differential expression at the transcription level between experimental samples. However, tailored DTE methods are still under development. The difficulty of DTE analysis is the

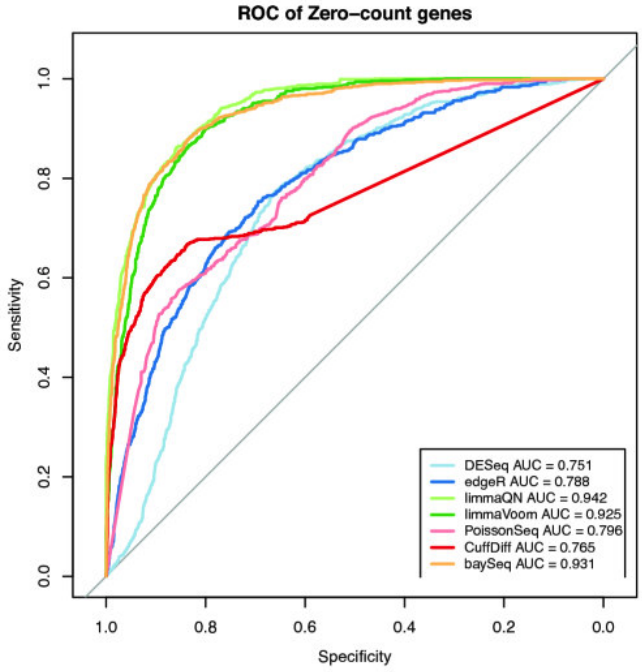

Figure 4.: (adapted from [38])ROC curves for detection of DE.

uncertainty introduced by the process of estimating transcript abundance. The uncertainty is caused by reads that align to shared subsequences of transcripts, since it is not possible to tell which transcript they originate from. The effect of the uncertainty for DTE analysis is not well understood. A naive example is presented in Figure 5. The statistical evidence for differential expression differs considerably when uncertainty of estimation is taken into account. For instance, the variability of the estimate of the fold change for the high uncertainty case should be much larger than for the low uncertainty case. This variability may be present in a certain proportion of the significance for testing differential expression of this transcript. Currently, only a handful of methods can handle the uncertainty of transcript abundance estimation, including ***Cuffdiff2*** [52], ***BitSeq*** [15], ***MetaDiff*** [21] and ***Sleuth*** [37]. However, some methods are computationally intensive or their performance for real datasets lack complete examination by existing gold standards.

1.6.2 Challenges for estimating transcript abundance

The major challenge for estimating transcript abundance is the assignment of reads ambiguously mapped to multiple transcripts of a gene with many complex isoforms. Current efforts by the bioinformatics community to solve this problem are to process the estimation in a probabilistic manner. Common procedures (used in ***Cufflinks2*** [52]) and ***RSEM*** [26]), can be summarized in the following steps: building models that probabilistically assigns aligned reads to gene isoforms (i.e., setting up a probabilistic model); calculating the fraction of a read that is derived from each transcript using specific algorithms (e.g., the expectation-maximization (EM) algorithm or Markov chain Monte Carlo (MCMC) algorithm); and finally, providing the results as an expected count-table; optionally, reporting the errors in estimation of transcript abundance (e.g., ***Cufflinks2*** reports the standard errors based on importance sampling methods). However, the methods based on this strategy have several issues. For

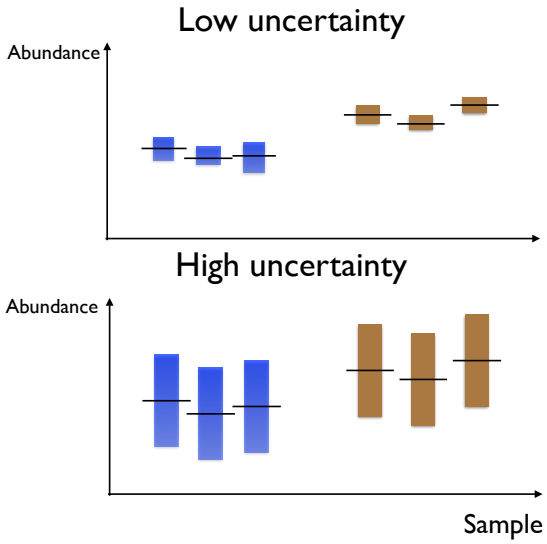

Figure 5.: Schematic illustration of the effect of uncertainty of transcript abundance estimation for DTE analysis in a case of a two group comparison (3 versus 3). The variability of estimation of the fold change for the high uncertainty case is much larger than for the low uncertainty case.

Cufflinks2, its incomplete usage of the EM algorithm would cause a loss of estimation accuracy for multi-mapping reads (i.e., the strategy of ***Cufflinks2*** to resolve ambiguous mapping reads was considered as one iteration of the EM algorithm of ***RSEM*** [26]). ***RSEM*** can achieve a high accuracy but its approach based on the EM algorithm is computationally intensive. For ***BitSeq***, its Bayesian approach of using the MCMC algorithm for estimation of transcript abundances is also computationally intensive. Recently new methods were designed to directly estimate transcript abundance from raw reads without relying on alignments, so-called alignment-free methods (e.g., ***Sailfish*** [34], ***Salmon*** [33] and ***kallisto*** [6]). They successfully solved computational bottlenecks during estimation of transcript abundance; this allows bootstrapping of reads with replacement and re-estimation within a reasonable amount of time. Using the bootstraps, the uncertainty of estimation can be quantified and the next challenge is to propagate this to the DTE analysis.

1.6.3 Current methods for DTE analysis

There are various tools for DTE analysis. Many tools provide a combined solution that includes transcript abundance estimation and DE analysis (e.g., ***kallisto*** plus ***Sleuth***, ***RSEM*** plus ***EBSeq*** , ***Cufflinks2*** plus ***Cuffdiff2*** and ***BitSeq***). Here focusing on the detection of differentially expressed transcripts, methods can be basically classified into two approaches: i) directly using count-based methods such as ***edgeR*** given a transcript-abundance-table (i.e., ignoring uncertainty in estimation) and ii) specific statistical models that can handle the uncertainty of transcript abundance estimation, but require pre-computed technical variance of transcript abundance provided by quantitation tools (e.g., ***MetaDiff*** requires estimated standard errors of transcript abundance from ***Cufflinks*** [53].).

The first approach is considered an extension of detecting DE genes given a transcript-abundance-table instead of a gene-abundance-table. This straightforward strategy (e.g., ignoring uncertainty in estimation) can directly inherit many advantages from the count-based methods. For instance, ***edgeR***, primarily designed to focus on differences at the gene level, contains the

following features for count data: the GLM framework allowing a flexible design matrix, the NB model naturally fitting overdispersed data well, and moderated dispersion estimation towards a trend-by-mean by maximizing adjusted Cox-Reid profile likelihood (APL) increasing the power of estimation. This approach has been shown to give good performances across a range of simulated and experimental datasets [47].

The second approach is to treat the transcript abundance estimation error as a measurement error in a model framework (e.g., ***Sleuth***, ***MetaDiff*** and ***MMSEQ*** [54]). For ***Sleuth***, the estimated variance of transcript abundance comes from bootstrapping samples. ***MetaDiff*** approximates (asymptotic) variances of transcript abundance from ***Cufflinks*** using the delta method and borrows ideas from meta-regression to handle different sources of variability: within- and between-study variation, allowing adjustment of a variance component between variabilities in isoform expression estimation (within-study) and variations in isoform expression levels across samples (between-study). However, this approach lacks an appropriate shrinkage or moderation procedure to improve variance (or dispersion) estimation accuracy. ***Sleuth*** makes use of the bootstrap to estimate technical variation in experiments and passes them to a response error measurement model to adjust for the uncertainty of transcript abundance estimation. ***MMSEQ***, based on a Bayesian model, summarizes the uncertainty from the posterior distribution of each expression parameter. ***EBSeq*** firstly partitions whole transcripts into groups according to isoform complexity then estimates the model parameters of each subgroup independently using an empirical Bayesian model [25]. ***Cuffdiff2*** based on mixing NB distributions can handle not only biological and technical variation from sample replicates but also the uncertainty of transcript abundance estimation. Methods that can handle the uncertainty of estimation are theoretically superior. However, the performance of these methods on real data is unknown.

2 Research objectives and challenges

2.1 Research objectives

2.2 Objective 1: a robust statistical method for detecting differential expression in RNA-seq data

Building a robust method for DE analysis was the core task of my PhD studies:

1. Investigate the patterns of outliers existing in real RNA-seq data (e.g., their identification and frequency).
2. Investigate the influence of outliers on model inference.
3. Build an open simulation framework that reflects as best as possible the reality of RNA-seq count data (with and without outliers).
    - Characterize properties of real data. For instance, both in terms of how many genes that should be differential expressed and what should be the relative expressed level? This can be learned from existing datasets.
    - Which model could represent the reality of the RNA-seq data best? Should be the classical parametric model, such as the negative binomial model, or a non-parametric model, such as plasmode model [39] or any others?
    - Formalize outlier mechanism in simulation framework. What is the methodology

to add an outlier?

4. Build a robust model framework that can dampen the effect of outliers and comparing its performance agains current existing methods.
    - Choose a good strategy to dampen the effect of outliers (e.g., weighted likelihood or robust M-estimator), which can be possibly adopted into current ***edgeR*** model framework.
    - Make comparison to existing methods (i.e., benchmark).

Although we built a robust method and a simulation framework, we still made some efforts to make DE method more comparison convenient, reproducible and visualizable through the use of web-based applications.

2.3 Objective 2: an open framework based on R code for benchmarking genome-scale methods

I will focus on the following points for this project:

1. Make a simple container, which can store p-value, labels and other necessary values for quantifying performance of DE methods.
2. Build customized metrics, such as partial ROC curves and power curves, providing a flexible, visual and correct benchmark result to evaluate performance of selected DE methods.

2.4 Objective 3: zero-counts

For this, I mainly focus on:

1. Investigating the reality of zero-count for real datasets. How many genes contains zero-counts in a real RNA-seq dataset?
2. Researching the global effect of zero-counts on moderated dispersion estimates. Should the moderation make the results worse when faced with zero-counts?
3. Identifying mechanisms that can identify this pattern.

2.5 Objective 4: DTE analysis

The primary goals of this project are the following:

1. As a baseline, investigate the feasibility using ***edgeR*** (and similar tools) with RNA-seq quantitation methods in DTE analysis (e.g, ignoring uncertainty).
2. Investigate additional models, such as random effects, weighting, bootstrapping etc for the propagation of abundance estimation uncertainty.
3. Build a complete, extensible and open simulation framework for the pipeline of DTE analysis.
    - Read-based simulation
    - Large parameter space of simulation settings (e.g., number of replicates, genes and transcripts)
    - The NB model
    - Empirical parameters generated from real data
    - Read errors from real data
4. Tailored metrics for benchmarking DTE methods.

5. Make best-practice recommendations for the community.

2.6 Research challenges

2.6.1 Limitation of existing model framework

The current moderation procedure of ***edgeR*** that can shrink dispersion estimates towards a common trend via a weighted APL framework (as mention in Section 4) is developed because there is no conjugate prior of NB model. This framework plays the role of an approximate empirical Bayesian model, which improves the reliability and accuracy of estimation with limited replicates. Since it has been successful over several years, new methodology should inherit from the existing moderation framework, or else the gain of new method may not be compensate for the loss of giving up the existing framework. However, current framework is difficult to directly add extensions to. For instance, mixed NB model without a close form of marginal likelihood requiring a certain approximation (e.g., Laplace) is difficult to implement into this framework. If it was possible, the computational time would not be acceptable. Additionally, for developing robust methods, the robust M-estimator that is conventionally useful for robust methods, is difficult to add into this framework. This is the motivation we developed robust method based weighted likelihood estimation using observation weights.

2.6.2 Errors from (external) quantification tools

For DTE analysis, bootstrapping sample (BS) counts generated by some quantification tools (e.g., ***kallisto***) would be useful for handling the uncertainty of transcript abundance estimation. However, we are concerned that current quantification tools may produce the errors into BS counts during the bootstraps. Practically, we found that technical replicates of inferred counts (counts within one BS) with zero uncertainty do not follow a Poisson distribution. If BS counts are propagated to DTE model (see Section 2.1), a potential loss would be caused by the errors particularly in case of datasets with low uncertainty. In a certain point of view, the process of bootstraps by quantification tool seems like a black box: currently, its internal process could not be traced. More efforts (e.g., providing more options and more information of BS) on the quantification tools would be necessary for developing tailored DTE methods.

3 Thesis Summary

This thesis consists of five chapters: four papers and one chapter of discussion and perspective. Their content and contribution are briefly summarized below.

Chapter I

Robustly detecting differential expression in RNA sequencing data using observation weights

by Xiaobei Zhou, Helen Lindsay and Mark D. Robinson [58]

This paper introduces a new robust approach that can dampen the effect of outliers in RNA-seq within the existing GLM/moderated-dispersion-to-trend framework. It is available in

the current version of R/Bioconductor ***edgeR*** package. In addition, a prototype of an extensible simulation framework is developed for generating count tables reflective of real data to facilitate comprehensive testing of current and future methods. The robust method employs strategies from classical robust statistics using a weight-and-re-estimate methodology according to some definition of model fit (e.g., Pearson residual or Deviance residual) and a score that controls the influence of each observation (e.g., Huber function). The idea is to attach a weight to each observation; observations that deviate strongly from the model fit are given lower weight in the next iteration of estimation. In particular, Pearson residuals from the current fit are sent through a weight function, which gets passed to the next iteration of estimation. The dispersion estimation machinery (i.e., trended APL) also receives the same observation weight, so that the influence of outliers is dampened on both the regression and dispersion estimates. This method achieves the desired result: resistance to outliers while maintaining high power.

Chapter II

benchmarkR: an R package for benchmarking genome-scale methods

by Xiaobei Zhou, Charity W. Law and Mark D. Robinson [57]

In this paper, we introduce ***benchmarkR***, an R package designed to assess and visualize the performance of statistical methods for datasets that have an independent truth. This package provides several standard metric plots with customized modifications, such as "rocX" (ROC plot with an X point marking the location of the method's FDR). The main innovation of the package is a flexible power-versus-achieved FDR plot providing a visualized solution to observe the trend or tradeoff between power detection and FDR control of evaluated methods. Additionally, this metric can provide information on how well the methods are calibrated (i.e., whether they achieve their expected FDR control). The benchmarkR package is available from: `https://github.com/markrobinsonuzh/benchmarkR` with a detailed vignette. The main manuscript is currently under review.

Chapter III

Do count-based differential expression methods perform poorly when genes are expressed in only one condition?

by Xiaobei Zhou and Mark D. Robinson [60]

Genes expressed in only one condition occur in almost all RNA-seq experiments. A report in Genome Biology from Rapaport and co-authors claimed that count-based methods (e.g., ***edgeR***) drop in power when genes are expressed in only one condition [38]. This result drew our interest to understand how aspects of all-zero-in-one-condition manifest undesirable properties in count-based models (e.g., estimates on the boundary of the parameter space). Through further analysis of this result, several findings are pointed out in a Correspondence article: i) Rapaport et al. made an error in the signal-to-noise (S/N) calculation for ***edgeR***. ii) A customized simulation suggests that count-based methods perform as well or better than other methods, counter to the original conclusion. iii) We add to the discussion about how various choices in benchmarking affect the results.

Chapter IV

miRNA-Seq normalization comparisons need improvement

by Xiaobei Zhou, Alicia Oshlack and Mark D. Robinson [59]

This paper is a Letter to the Editor referring to "Evaluation of normalization methods in mammalian microRNA-Seq data" by Garmire and Subramaniam [13]. Garmire et al. comprehensively evaluated several normalization methods for microRNA sequencing data (miRNA-Seq) with partial truth labels from quantitative PCR results. The comparison contains the methods currently popular for messenger RNA sequencing (mRNA-Seq) data, such as total-depth normalization ("raw") and Trimmed Mean of M-values ("TMM"). Additionally, several methods that are commonly used for biological data, such as global scaling and quantile normalization (QN) are also included. Their conclusion of poor performance and "abnormal results" of the TMM method motivated us to explore the situation of how well the normalization methods primary designed for mRNA-Seq can apply to miRNA-Seq. After investigating, we presented reproducible reanalyses to claim that poor performance of TMM was the result of a coding error that shifted log-ratios in the wrong direction. Furthermore, we pointed out that various practical issues, such as the sensitivities of ROC results when there is a limited number of true labels (see Supplementary Note), were not satisfyingly discussed.

Chapter V

Discussion and perspectives

This chapter discusses the issues of our robust method in ***edgeR***, including the robust estimator selection and (potential) relationship between poor false discovery control and inappropriate weights used in our robust ***edgeR*** method. In addition, my early stage research on differential transcript expression analysis are discussed as future perspectives. Several model frameworks developed for incorporating bootstrapping sample counts provided by quantification tool for differential transcript expression analysis are described here.

References

[1] Gael P Alamancos, Eneritz Agirre, and Eduardo Eyras. Methods to study splicing from high-throughput RNA sequencing data. *Methods Mol Biol*, 1126:357–97, jan 2014.

[2] Simon Anders. HTSeq: Analysing high-throughput sequencing data with Python.

[3] Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Genome biology*, 11(10):R106, jan 2010.

[4] Simon Anders, Alejandro Reyes, and Wolfgang Huber. Detecting differential usage of exons from RNA-seq data. *Genome Res*, 22(10):2008–17, 2012.

[5] Bradley E Bernstein, Ewan Birney, Ian Dunham, Eric D Green, Chris Gunter, and Michael Snyder. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, 2012.

[6] Nicolas Bray, Harold Pimentel, Páll Melsted, and Lior Pachter. Near-optimal RNA-Seq quantification. *ArXiv e-prints*, arXiv, may 2015.

[7] Tiffany J Chen and Nikesh Kotecha. Cytobank: providing an analytics platform for community cytometry data analysis and collaboration. *Current topics in microbiology and immunology*, 377:127–57, 2014.

[8] D R Cox and N Reid. Parameter Orthogonality and Approximate Conditional Inference. *Journal of the Royal Statistical Society Series B Methodological*, 49(1):1–39, 1987.

[9] Francis Crick and Others. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.

[10] Alexander Dobin, Carrie A Davis, Felix Schlesinger, Jorg Drenkow, Chris Zaleski, Sonali Jha, Philippe Batut, Mark Chaisson, and Thomas R Gingeras. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 29(1):bts635–, 2013.

[11] Bradley Efron, Robert Tibshirani, John D Storey, and Virginia Tusher. Empirical Bayes Analysis of a Microarray Experiment. *Journal of the American Statistical Association*, 96(456):1151–1160, 2001.

[12] Pär G Engström, Tamara Steijger, Botond Sipos, Gregory R Grant, André Kahles, Gunnar Rätsch, Nick Goldman, Tim J Hubbard, Jennifer Harrow, Roderic Guigó, and Paul Bertone. Systematic evaluation of spliced alignment programs for RNA-seq data. *Nature methods*, 10(12):1185–91, 2013.

[13] Lana Xia Garmire and Shankar Subramaniam. Evaluation of normalization methods in mammalian microRNA-Seq data. *RNA*, 18(6):rna.030916.111–, 2012.

[14] Robert C Gentleman, Vincent J Carey, Douglas M Bates, Ben Bolstad, Marcel Dettling, Sandrine Dudoit, Byron Ellis, Laurent Gautier, Yongchao Ge, Jeff Gentry, Kurt Hornik, Torsten Hothorn, Wolfgang Huber, Stefano Iacus, Rafael Irizarry, Friedrich Leisch, Cheng Li, Martin Maechler, Anthony J Rossini, Gunther Sawitzki, Colin Smith, Gordon Smyth, Luke Tierney, Jean YH Yang, and Jianhua Zhang. Bioconductor: open software development for computational biology and bioinformatics. *Genome Biology*, 5(10):R80, 2004.

[15] Peter Glaus, Antti Honkela, Magnus Rattray, Glaus, Peter, Honkela, Antti, Rattray, and Magnus. Identifying differentially expressed transcripts from RNA-seq data with biological variation. *Bioinformatics*, 28(13):1721–1728, 2012.

[16] Manfred G Grabherr, Brian J Haas, Moran Yassour, Joshua Z Levin, Dawn A Thompson, Ido Amit, Xian Adiconis, Lin Fan, Raktima Raychowdhury, Qiandong Zeng, Zehua Chen, Evan Mauceli, Nir Hacohen, Andreas Gnirke, Nicholas Rhind, Federica Di Palma, Bruce W Birren, Chad Nusbaum, Kerstin Lindblad-Toh, Nir Friedman, and Aviv Regev. Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature Biotechnology*, 29(7):644–652, 2011.

[17] Frank E Grubbs. Procedures for detecting outlying observations in samples. *Technometrics*, 11(1):1–21, 1969.

[18] Kasper D Hansen, Steven E Brenner, and Sandrine Dudoit. Biases in Illumina transcriptome sequencing caused by random hexamer priming. *Nucleic Acids Research*, 38(12):e131, 2010.

[19] Kasper D Hansen, Rafael A Irizarry, and Zhijin Wu. Removing technical variability in RNA-seq data using conditional quantile normalization. *Biostatistics*, 13(2):204–216, 2012.

[20] Thomas J Hardcastle and Krystyna A Kelly. baySeq: Empirical Bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, 11(1):422, 2010.

[21] Cheng Jia, Weihua Guan, Amy Yang, Rui Xiao, W. H. Wilson Tang, Christine S. Moravec, Kenneth B. Margulies, Thomas P. Cappola, Mingyao Li, and Chun Li. MetaDiff: differential isoform expression analysis using random-effects meta-regression. *BMC Bioinformatics*, 16(1):208, jul 2015.

[22] Daehwan Kim, Geo Pertea, Cole Trapnell, Harold Pimentel, Ryan Kelley, and Steven L Salzberg. TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome biology*, 14(4):R36, 2013.

[23] Ben Langmead, Cole Trapnell, Mihai Pop, and Steven L Salzberg. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome biology*, 10(3):R25, jan 2009.

[24] Charity W Law, Yunshun Chen, Wei Shi, and Gordon K Smyth. Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biology*, 15(2):R29, 2014.

[25] Ning Leng, John A Dawson, James A Thomson, Victor Ruotti, Anna I Rissman, Bart M G Smits, Jill D Haag, Michael N Gould, Ron M Stewart, and Christina Kendziorski. EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics Oxford England*, 29(8):1035–1043, 2013.

[26] Bo Li and Colin N Dewey. RSEM: accurate transcript quantification from RNA-seq data with or without a reference genome. *BMC Bioinform*, 12(1):323, 2011.

[27] H Li, J Ruan, and R Durbin. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Res.*, 18(11):1851–1858, nov 2008.

[28] Jun Li and Robert Tibshirani. Finding consistent patterns: A nonparametric approach for identifying differential expression in RNA-Seq data. *Statistical Methods in Medical Research*, 22(5):519–39, 2013.

[29] Yang Liao, Gordon K Smyth, and Wei Shi. FeatureCounts: an efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics*, 30(7):923–30, apr 2014.

[30] Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for RNA-Seq data with DESeq2. *Genome Biol*, 15(12):550, feb 2014.

[31] Davis J. McCarthy, Yunshun Chen, and Gordon K. Smyth. Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic acids research*, 40(10):1–10, jan 2012.

[32] Guilherme Neves, Jacob Zucker, Mark Daly, and Andrew Chess. Stochastic yet biased expression of multiple Dscam splice variants by individual cells. *Nature genetics*, 36(3):240–246, 2004.

[33] Rob Patro, Geet Duggal, and Carl Kingsford. Salmon: Accurate, Versatile and Ultrafast Quantification from RNA-seq Data using Lightweight-Alignment. *bioRxiv*, 2015.

[34] Rob Patro, Stephen M Mount, and Carl Kingsford. Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nature biotechnology*, 32(5):462–464, 2014.

[35] Belinda Phipson, Stanley Lee, Ian J. Majewski, Warren S. Alexander, and Gordon K. Smyth. Robust hyperparameter estimation protects against hypervariable genes and improves power to detect differential expression. 10:1–23, 2016.

[36] Joseph K Pickrell, John C Marioni, Athma A Pai, Jacob F Degner, Barbara E Engelhardt, Everlyne Nkadori, Jean-Baptiste Veyrieras, Matthew Stephens, Yoav Gilad, and Jonathan K Pritchard. Understanding mechanisms underlying human gene expression variation with RNA sequencing. *Nature*, 464(7289):768–772, apr 2010.

[37] Harold J Pimentel, Nicolas Bray, Suzette Puente, Páll Melsted, and Lior Pachter. Differential analysis of RNA-Seq incorporating quantification uncertainty. *bioRxiv*, 2016.

[38] Franck Rapaport, Raya Khanin, Yupu Liang, Mono Pirun, Azra Krek, Paul Zumbo, Christopher E Mason, Nicholas D Socci, and Doron Betel. Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data. *Genome Biol*, 14(9):R95, 2013.

[39] Pablo Reeb and Juan Steibel. Evaluating statistical analysis models for RNA sequencing experiments. *Frontiers in Genetics*, 4:178, 2013.

[40] Alejandro Reyes, Simon Anders, and Wolfgang Huber. Inferring differential exon usage in RNA-Seq data with the DEXSeq package. *Bioconductor vignettes*.

[41] Adam Roberts, Harold Pimentel, Cole Trapnell, and Lior Pachter. Identification of novel transcripts in annotated genomes using RNA-Seq. *Bioinformatics*, 27(17):2325–9, sep 2011.

[42] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, jan 2010.

[43] Mark D Robinson and Gordon K Smyth. Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23(21):2881–2887, 2007.

[44] Mark D Robinson and Gordon K Smyth. Small-sample estimation of negative binomial dispersion, with applications to SAGE data. *Biostatistics Oxford England*, 9(2):321–332, 2008.

[45] Gordon K Smyth. Limma : Linear Models for Microarray Data. *Bioinformatics*, pages(2005):397–420, 2005.

[46] Charlotte Soneson and Mauro Delorenzi. A comparison of methods for differential expression analysis of RNA-seq data. *BMC Bioinformatics*, 14(1):91, jan 2013.

[47] Charlotte Soneson, Michael I Love, and Mark D Robinson. Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences. *F1000Research*, 4:1521, jan 2015.

[48] Charlotte Soneson, Katarina L Matthes, Malgorzata Nowicka, Charity W Law, and Mark D Robinson. Isoform prefiltering improves performance of count-based methods for analysis of differential transcript usage. *Genome Biology*, 17(1):1–15, 2016.

[49] Sonia Tarazona, Fernando García-Alcalde, Joaquín Dopazo, Alberto Ferrer, and Ana Conesa. Differential expression in RNA-seq: A matter of depth. *Genome Research*, 21(12):2213–2223, 2011.

[50] Cole Trapnell, David G Hendrickson, Martin Sauvageau, Loyal Goff, John L Rinn, and Lior Pachter. Differential analysis of gene regulation at transcript resolution with RNA-seq. *Nat Biotechnol*, 31(1):46–53, jan 2012.

[51] Cole Trapnell, Lior Pachter, and Steven L Salzberg. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 25(9):1105–11, may 2009.

[52] Cole Trapnell, Adam Roberts, Loyal Goff, Geo Pertea, Daehwan Kim, David R Kelley, Harold Pimentel, Steven L Salzberg, John L Rinn, and Lior Pachter. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat Protoc*, 7(3):562–78, mar 2012.

[53] Cole Trapnell, Brian A Williams, Geo Pertea, Ali Mortazavi, Gordon Kwan, Marijke J Van Baren, Steven L Salzberg, Barbara J Wold, and Lior Pachter. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat Biotechnol*, 28(5):511–5, 2010.

[54] Ernest Turro, Shu-Yi Su, Angela Goncalves, Lachlan J M Coin, Sylvia Richardson, and Alex Lewin. Haplotype and isoform specific expression estimation using multi-mapping RNA-seq reads. *Genome Biology*, 12(2):R13, 2011.

[55] Zhong Wang, Mark Gerstein, and Michael Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63, 2009.

[56] Dirk O. Watermann, Yesheng Tang, Axel Zur Hausen, Markus Jäger, Stefan Stamm, and Elmar Stickeler. Splicing factor Tra2-β1 is specifically induced in breast cancer and regulates alternative splicing of the CD44 gene. *Cancer Research*, 66(9):4774–4780, 2006.

[57] Xiaobei Zhou, Charity W Law, and Mark D Robinson. benchmarkR: an R package for benchmarking genome-scale methods. Technical report, apr 2015.

[58] Xiaobei Zhou, Helen Lindsay, and Mark D Robinson. Robustly detecting differential expression in RNA sequencing data using observation weights. *Nucleic Acids Research*, 42(11):e91, jan 2014.

[59] Xiaobei Zhou, Alicia Oshlack, and Mark D Robinson. miRNA-Seq normalization comparisons need improvement. *RNA (New York, N.Y.)*, 19(6):733–4, 2013.

[60] Xiaobei Zhou and Mark D Robinson. Do count-based differential expression methods perform poorly when genes are expressed in only one condition? *Genome biology*, 16:222, jan 2015.

CHAPTER I

Robustly detecting differential expression in RNA sequencing data using observation weights

Xiaobei Zhou, Helen Lindsay and Mark D. Robinson

Paper published in *Nucleic Acids Research (2014), 42, pp. e91*

Nucleic Acids Research Advance Access published April 20, 2014

doi: 10.1093/nar/gku310

Robustly detecting differential expression in RNA sequencing data using observation weights

Xiaobei Zhou[1,2], Helen Lindsay[1,2] and Mark D. Robinson[1,2,*]

[1]Institute of Molecular Life Sciences, University of Zurich, CH-8057 Zurich, Switzerland and [2]SIB Swiss Institute of Bioinformatics, University of Zurich, CH-8057 Zurich, Switzerland

Received December 4, 2013; Revised March 10, 2014; Accepted March 31, 2014

ABSTRACT

A popular approach for comparing gene expression levels between (replicated) conditions of RNA sequencing data relies on counting reads that map to features of interest. Within such count-based methods, many flexible and advanced statistical approaches now exist and offer the ability to adjust for covariates (e.g. batch effects). Often, these methods include some sort of 'sharing of information' across features to improve inferences in small samples. It is important to achieve an appropriate trade-off between statistical power and protection against outliers. Here, we study the robustness of existing approaches for count-based differential expression analysis and propose a new strategy based on observation weights that can be used within existing frameworks. The results suggest that outliers can have a global effect on differential analyses. We demonstrate the effectiveness of our new approach with real data and simulated data that reflects properties of real datasets (e.g. dispersion-mean trend) and develop an extensible framework for comprehensive testing of current and future methods. In addition, we explore the origin of such outliers, in some cases highlighting additional biological or technical factors within the experiment. Further details can be downloaded from the project website: http://imlspenticton.uzh.ch/robinson_lab/edgeR_robust/.

INTRODUCTION

RNA sequencing (RNA-seq) is widely used for numerous biological applications, including the detection of alternative splice forms, ribonucleicacid (RNA) editing, allele-specific expression profiling, novel transcript discovery but most commonly, for detecting changes in expression between experimental conditions or treatments. Compared to microarray technology, RNA-seq offers an open system, higher resolution, lower relative cost and less bias (1). A typical RNA-seq experiment includes: (i) capture of an RNA subpopulation (e.g. polyA-enriched, depleted of ribosomal ribonucleicacid) from cells of interest; (ii) reverse transcription into complementary DNA (cDNA); (iii) preparation and sequencing of millions of short cDNA fragments (∼200 bp); (iv) mapping to a reference genome or (assembled) transcriptome; (v) counting according to a catalog of features. This last counting step can be conducted by excluding ambiguous reads between genes (2), or with advanced tools that portion ambiguous reads to transcripts (3) or can be done in combination with assembly tools (4). The focus here is on methods for count-based differential expression (DE) analyses and the robustness thereof; thus, the starting point here is a count table of features-by-samples, such as those available from the ReCount project (5).

Considerable recent effort has been paid by the statistical community to the discovery of DE features, given a count table; recent comparisons have shown that no method dominates the spectrum of possible situations (6,7). RNA-seq remains expensive and in many cases researchers are studying precious samples or rare cell types, so the number of biological replicates is often limiting. It is clear that the most successful methods implement some form of 'information sharing' across the whole dataset to improve DE inference (2), and this becomes an intricate exercise to trade-off power, false discovery control and protection against outliers. To highlight this distinction, we describe two popular software implementations for the negative binomial (NB) model, which arguably is the *de facto* standard for accounting for biological variability in such genome-scale count datasets. The latest version of edgeR moderates dispersion estimates toward a trended-by-mean estimate (8), whereas DESeq takes the maximum of a fitted dispersion-mean trend or the individual feature-wise dispersion estimate (9). The effect imposed on features with 'outliers' is illustrated in Figure 1. Ten randomly selected samples from individuals from the HapMap project (denoted as Pickrell (10)) are divided into two groups of 5, forming an artificial 'null' scenario. While very little true differential expression is expected, a low rate of false detections occur; in particular, edgeR detects a small number of genes with low estimated false discovery rate that exhibit one or two observa-

*To whom correspondence should be addressed. Tel: +41 44 635 48 48; Fax: +41 44 635 68 68; Email: mark.robinson@imls.uzh.ch

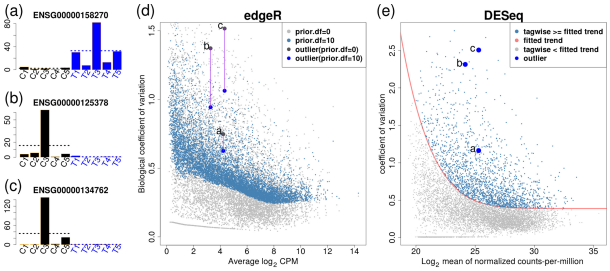

Figure 1. From Pickrell (10) data, 10 randomly selected samples from individuals are divided into two groups of 5, forming an artificial 'null' scenario. (**a**), (**b**) and (**c**) show barplots of log-counts-per-million (CPMs) of three genes from the top 10 DE genes with one or two extremely large observations. Dashed lines represent group-wise average log-CPMs. (**d**) and (**e**) plot genewise biological coefficient of variation (BCV) against gene abundance (in $\log_2$ counts per million) for edgeR and DESeq. In panel (d), gray dots show unmoderated biological BCV estimates ($\sqrt{\phi_i}\sqrt{\phi_i}$) (prior degrees of freedom = 0). Steel blue dots show moderated biological BCV with prior degree 10 (default setting for edgeR). Three outlier genes on (a), (b) and (c) are labeled by large blue dots. For (e), DESeq uses the maximum (steel blue dots) of a fitted dispersion-mean trend (red line) or the individual feature-wise (tagwise) dispersion estimate. Three outlier genes are also pointed out by large blue dots.

tions that are generally much higher in expression (Figure 1a–c). We believe that there are two causes for this: (i) the sensitivity of relative expression estimates to these 'outlying' observations; (ii) moderation of the dispersion estimates toward the trend. In contrast, DESeq remains largely unaffected by these outliers, since the dispersion estimation policy is to keep the maximum; in what follows, we will explore the effect of this maximum policy on power. All computed statistics for this dataset are stored in Supplementary Table S1.

The downstream effect of these dispersion estimation strategies suggest: (i) DESeq is generally conservative but robust; (ii) edgeR can be sensitive to outliers when there is sufficient dispersion smoothing toward the trend (effectively underestimating the dispersion in the shrinking process), but should be more powerful in the absence of such extreme observations (2). Our goal in the current study is to achieve a suitable middle ground, perhaps forfeiting a small amount in statistical efficiency, similar to established robustness frameworks, to reduce the influence of extreme observations in differential expression calls. As hinted above and in general, robustness is not solely determined by the dispersion parameter, but also by controlling the influence of outliers to other parameters in the model (e.g. those representing changes in expression). We explore these aspects in both simulated and real data, provide a extensible framework for evaluating the tradeoffs and highlight some instances of biology or technical factors that may give outliers.

The literature is rich in alternatives for count-based DE analyses and in particular, dispersion estimation, yet it remains increasingly difficult to assess the performance across the range of possibilities. For example, recent evidence suggests that one can suitably transform count data and analyze with methods developed for microarrays, with special treatment (11). The mainstream strategy is to directly fit count data to extensions of the Poisson model and in particular, the NB model. Many implementations are available as R/Bioconductor packages (12), such as edgeR (13), DESeq (9), ShrinkBayes (14), baySeq (15) and variations of dispersion estimation that can be used within existing implementations (16); the main differences lie in the estimation of the dispersion or in the inference machinery (e.g. Bayesian versus frequentist). Recent comparisons and summaries of the methods available can be found in (2), (6) and (7).

Some early and existing count-based DE analysis tools only allowed two-group comparisons. That is, they could not handle more complex situations, such as paired samples, time courses or batch effects. Recently, McCarthy *et al*. developed generalized linear model (GLM) capabilities in edgeR (8), allowing a much broader class of experimental designs to be analyzed and other frameworks have followed suit. However, GLMs require iterative fitting and more complicated dispersion estimation machinery (8). As shown in Figure 1, this framework can suffer a lack of robustness, whereby even a single extreme value (outlier) could largely affect estimates of regression parameters (e.g. mean of experimental condition), as highlighted by recent comparative studies (6) (see also Figure 1). In addition, the moderation of the dispersion parameter toward a trended value is actually contributing to the lack of robustness, forcing the dispersion to be underestimated (Figure 1). DESeq2 (successor of DESeq) takes an altogether different stance on robustness: using a Cook's distance metric, features that exhibit an extreme value are not considered for downstream statistical testing.

The strategy proposed in this paper is that of 'observation weights', effectively down-weighting outliers to dampen their influence. There is already some precedent for doing this in GLM settings: Carroll and Pederson (17) introduced weighted maximum likelihood estimators for the logistic model; Cantoni (18) presented a robust quasi-likelihood approach for inference in binomial and Poisson models; Agostinelli and Alqallaf (19) derived weighted likelihood equation for GLMs by directly inserting 'observation weights' into iterative re-weighted least squares algorithm (IRLS). Of particular importance, after adding observation weights, the asymptotic theory suggests that likelihood ratio statistics of model parameters still converge to approximate chi-squared distributions under the null hypothesis (20). At present, no 'off-the-shelf' robust approach is readily available for the negative binomial model in the context of genome-scale computations. In this paper, we build an outlier-resistant framework that maintains high power and achieves decent false discovery control and make it available in the edgeR software package; the same strategy could be employed in other frameworks. We benchmark its performance on real and simulated data and explore the origins of outlying observations.

MATERIALS AND METHODS

A standard setup of NB model in GLM framework

To most easily explain the addition of observation weights, we follow closely the notation used in McCarthy *et al*. (8). Let the Y_{gi} be the read count in sample i for feature g ($g = 1, \ldots, G$). Assume Y_{gi} follows a NB distribution with mean μ_{gi}

and dispersion ϕ_g, denoted by $Y_{gi} \sim \mathrm{NB}(\mu_{gi}, \phi_g)$. Feature g's variance equals $\mu_{gi} + \phi_g \cdot {\mu_{gi}}^2$, while the dispersion ϕ_g represents the square of the 'biological coefficient of variation' (8). In the GLM setting, the mean response, μ_{gi}, is linked to a linear predictor, here with the canonical logarithm link according to:

$$\log(\mu_{gi}) = X\beta_g + \log N_i, \tag{1}$$

where X is the design matrix containing the covariates (e.g. experimental conditions, batch effects, etc.), β_g is a vector of regression parameters (a subset of which are of interest for differential expression inference) and N_i is the (effective) library size for sample i.

For estimation of the regression parameters, maximum likelihood estimation is used. The derivative of the log-likelihood, $l(\beta_g)$, with respect to the coefficient β_g is $X^T z z_g$, where $z z_{gi} = (y_{gi} - \mu_{gi})/(1 + \phi_g \mu_{gi})$. The estimated value of β_g can be obtained by the IRLS in the form:

$$\beta_g^{\mathrm{new}} = \beta_g^{\mathrm{old}} + (X^T \Omega_g X)^{-1} X^T z_g, \tag{2}$$

where $X^T \Omega_g X$ is the Fisher information matrix (also denoted below as $\mathcal{I}_g \mathcal{I}_g$) and Ω_g is the diagonal matrix of working weights, which are $\mu_g/(1 + \phi_g \mu_g)$ for the NB model.

Moderated and trended dispersion estimates

The adjusted profile likelihood (APL) introduced by Cox and Reid (21) has shown good performance for dispersion estimation in the context of genome-scale count data (8,22). The APL_g is a likelihood in terms of ϕ_g, penalized for the estimation of the regression parameters, β_g, as follows:

$$\mathrm{APL}_g(\phi_g) = \ell(\phi_g; \mathbf{y}_g, \hat{\beta}_g) - \frac{1}{2} \log |\mathcal{I}_g|, \tag{3}$$

where $\mathbf{y}_g \mathbf{y}_g$ is the vector of counts for gene g, $\hat{\beta}_g \hat{\beta}_g$ is the estimated coefficient vector, $\ell(\cdot)$ is the log-likelihood function, $\mathcal{I}_g \mathcal{I}_g$ is the Fisher information matrix and $|\cdot|$ is the determinant. The early strategy to accomplish moderation for the dispersion was by squeezing the tagwise dispersion toward a common dispersion that is estimated over all features (23). This weighted likelihood approach involves maximizing a linear weighting of the individual likelihood and the common (averaged) likelihood, the two terms, respectively, in

$$\arg\max \left\{ \mathrm{APL}_g(\phi_g) + \alpha \cdot \frac{1}{G} \sum_{k=1}^{G} \mathrm{APL}_k(\phi_g) \right\}, \tag{4}$$

where α is a suitably chosen weight.

A slight variation on this, which is now commonly applied after experience in many datasets showing a dispersion–mean relationship, is to shrink toward a dispersion estimated from features with similar average expression level (8). This so-called trended dispersion is constructed using local shared log-likelihood for feature g (more precisely, a smooth fit to common dispersions that are calculated in bins of averaged counts per million) and its neighboring features in terms of expression strength. Specifically, individual tagwise estimates for each feature can be estimated by maximizing a linearly weighted function between individual dispersion and local shared dispersion:

$$\hat{\phi}_g = \arg\max \left\{ \mathrm{APL}_g(\phi_g) + \gamma \cdot \mathrm{APL}_g^S(\phi_g) \right\}, \tag{5}$$

where $\hat{\phi}_g \hat{\phi}_g$ is moderated tagwise dispersion, γ is the prior degree of freedom afforded to the shared likelihood and

$$\mathrm{APL}_g^S(\phi_g) = \frac{1}{|C|} \sum_{k \in C} \mathrm{APL}_k(\phi_g), \tag{6}$$

where the set C represents features that are close to feature g in average log counts per million.

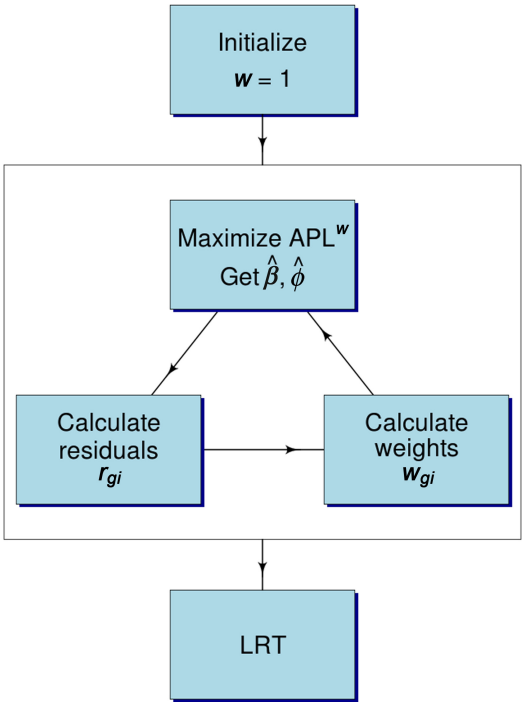

Figure 2. The flow chart of the robust algorithm implemented in edgeR. $\hat{\beta}\hat{\beta}$ is the estimated GLM regression coefficient and $\hat{\phi}\hat{\phi}$ is the moderated dispersion estimate by maximizing $\mathrm{APL}ww$ (Equation (10)). r_{gi} is the Pearson residual corresponding to count y_{gi} from Equation (7).ww_{gi} is the observation weight from Equation (8). LRT (glmLRT in edgeR) computes likelihood ratio tests using the weights.

A robust negative binomial GLM

Our approach to induce robustness is to attach a weight to each observation; observations that deviate strongly from the model fit are given lower weight. In particular, Pearson residuals from the current fit are sent through a weight function, which gets passed to the next iteration of estimation. The dispersion estimation machinery (i.e. trended APL) also receives the same observation weight, so that the influence of outliers is dampened on both the regression and dispersion estimates.The robust iterative estimation procedure using weights is described in Figure 2. The Pearson residual of an observed count y_{gi} from the NB GLM fit can

be calculated as

$$r_{gi} = \frac{y_{gi} - \hat{\mu}_{gi}}{\sqrt{\hat{\mu}_{gi}(1 + \hat{\phi}_g \hat{\mu}_{gi})}} \quad (7)$$

where $\hat{\mu}_{gi}\hat{\mu}_{gi}$ is the fitted value (from $\hat{\beta}\hat{\beta}$) and $\hat{\phi}_g\hat{\phi}_g$ is the moderated dispersion estimate. The Pearson residual is converted to weights using, e.g. the Huber function:

$$w_{gi} = w(r_{gi}) = \begin{cases} \frac{k}{abs(r_{gi})}, & \text{for abs}(r_{gi}) > k \\ 1, & \text{for abs}(r_{gi}) \leq k \end{cases} \quad (8)$$

where k represents a tuning constant for Huber estimator and is usually set to 1.345 in normally distributed settings to achieve 95% efficiency (24). This weight, ww_{gi}, gets used in the next iteration of GLM fitting; the IRLS equation becomes:

$$\beta_g^{\text{W-new}} = \beta_g^{\text{W-old}} + (X^T[W_g\Omega_g]X)^{-1} X^T[W_g]z_g \quad (9)$$

where W_g is the diagonal matrix of observation weights for feature g. The Fisher information matrix with observation weight becomes $\mathcal{I}_g^W = X^T[W_g\Omega_g]X\mathcal{I}_g^W = X^T[W_g\Omega_g]X$. In this approach, the APL for dispersion ϕ_g with observation weights can be written as

$$\text{APL}_g^W(\phi_g) = \ell^W(\phi_g; \mathbf{y}_g, \hat{\beta}_g) - \frac{1}{2}\log|\mathcal{I}_g^W|, \quad (10)$$

where $\ell^W(\cdot) \equiv \sum_i ww_{gi}l(\cdot)$ is the weighted log-likelihood function and $\mathcal{I}_g^W\mathcal{I}_g^W$ is the Fisher information matrix with observation weights. Then, using these dispersion estimates, the regression parameters are estimated, again using the observation weights.

For users of edgeR, only a small change in the standard pipeline is required.

A simulation framework with parameters based on the joint distribution of mean and dispersion estimates from RNA-seq data

We built a simulation framework that aims to accurately reflect the reality of RNA sequencing data. In order to evaluate the performance of our robust method and other methods across a variety of reasonable conditions, we created several options:

(1) nTags: total number of features,
(2) group: factor containing the experimental conditions,
(3) pDiff: proportion of DE features,
(4) foldDiff: relative expression level of truly DE features,
(5) pUp: proportion of DE features that increase in expression,
(6) dataset: dataset to take model parameters from,
(7) pOutlier: proportion of outliers to introduce,
(8) outlierMech: outlier generation mechanism to use.

We generate true NB model parameters, μ and ϕ, using the joint distribution of estimates, $\hat{\mu}\hat{\mu}$ and $\hat{\phi}\hat{\phi}$, estimated using edgeR from real datasets, such as the published count tables at ReCount (5): Pickrell (10), Cheung *et al*. (25,26). In particular, the joint distribution preserves the dispersion–mean trend, which can vary from dataset to dataset. After the removal of extremely high dispersions and low means (analogous to typical recommended filters; see Supplementary Figure S1), the derived-from-real-data parameters are used to simulate the counts, from a NB distribution and optionally with true DE.

To test robustness, we add outliers to the simulated counts. Outliers are large values and can be produced by two different mechanisms (outlierMech): first, counts are multiplied by a random factor between 1.5 and 10, as employed by Soneson and Delorenzi (6), and includes both the 'simple' (S) and 'random' (R) method. In S, a gene is chosen at some probability to have a single outlier randomly added. In R, each observation can become an outlier with some probability. In the second mechanism, called 'model' (M), each observation can become an outlier with some probability and if so, is sampled from a second NB distribution with larger μ (original μ multiplied by random factor between 1.5 and 10); R and M methods induce the same overall outlier rate.

Recently, van de Wiel *et al*. modeled genome-scale count data as zero-inflated negative binomial model (ZINB), which seemed to explain some of the dispersion–mean relationship (4). We have not considered simulations from ZINB distributions, since they do not appear to explain all of the observed dispersion–mean relationship in the datasets that we tested (see Supplementary Figure S2).

Methods compared

We evaluated and compared several methods for DE analysis, including edgeR, edgeR-robust, limma-voom, DESeq-pool, DESeq-glm, DESeq2, baySeq, SAMseq (27), EBSeq (28) and ShrinkBayes; the performance evaluation system that we developed allows arbitrary additions (assuming they are implemented in R). limma-voom is an extension to DE analysis of RNA-seq count data from limma (11); it transforms the count data with special treatment given to fitting the mean–variance relationship. DESeq is tested as two separate methods: DESeq-pool is the default setting method to estimate the empirical dispersion from all the conditions with replicates; DESeq-glm fits models according to a design matrix and estimates dispersion by maximizing APL. edgeR, DESeq and DESeq2 differ in how the dispersion is estimated: edgeR moderates dispersion toward a trended estimate (8), edgeR-robust expands this with observation weights, DESeq takes the maximum of a fitted trend of dispersion or the individual feature-wise dispersion estimate (9). DESeq2 offers a zero-mean normal prior on the log-fold-changes for moderation and a proper moderation of dispersion estimates to a trended value, except when the feature exhibits variability much greater than other features at the same expression strength; for outlier protection, a Cook's distance is calculated and those features with an extreme value are not promoted to formal statistical testing i.e. *P*-values are set to NA; in our simulations, these *P*-values are set to 1 so as to not remove features. The default normalization method is also different among edgeR, DESeq and DESeq2. edgeR uses trimmed-mean-of-M-values (TMM) (29), while DESeq and DESeq2 use a relative-log-expression approach. SAMseq, a non-parametric method,

employs Wilcoxon rank-sum statistics to estimate false discovery rate (FDR) through sample permutations.

baySeq, EBSeq and ShrinkBayes use Bayesian inference. baySeq employs the NB model and assumes that samples can be classified as different groups by their treatment conditions; samples within the same group should follow the same distribution and share parameters. Using an empirical Bayes approach, baySeq estimates the posterior probability of the null state. ShrinkBayes introduces the ZINB and performs inference using integrated nested Laplace approximations (INLA) (30,31) and provides Bayesian FDR and local false discovery rate (lfdr) (32) estimates. Since the computational cost of ShrinkBayes is high, some comparisons are skipped. EBSeq is similar to baySeq, providing posterior probability of DE, as well as EE (equally expressed), based on a parametric mixture model. Compared with other methods tested here, EBSeq can also detect DE isoforms in EE features, yet this is not our primary question here.

Notably, new methods, or variations of existing ones can be easily added to our comparison framework, simply by providing a wrapper to an R function that contains the correct inputs (count table, grouping variable) and outputs (*P*-values). See Supplementary web site for details.

Comparison metrics

To test the performance of each DE method in the presence of outliers, we employ several standard metrics and plots: false discovery (FD) plots, receiver operating characteristic (ROC) curves, partial ROC curves and power curves. Power (TP) curves and (partial) ROC curves (i.e. up to a certain false positive rate) evaluate the ability to distinguish, through statistical evidence, DE and non-DE. FD procedures gauge the control of the expected proportion of incorrectly rejected null hypotheses (33). Another useful plot is the relationship between TP rate and achieved false discovery rate across multiple thresholds.

An open graphical tool and R code for re-analysis: evaluating DE analysis methods

One disadvantage of current method comparisons (e.g. (6,7)) and those that accompany every new method published, is that they are a snapshot in time. If new methods come along, the developer must demonstrate that their method is better, by some metric. This task is important but somewhat repetitive, because many of the same metrics, plots and simulation models are (re-)implemented. We endeavored to create a system for performing standardized simulation-based testing.

In addition, all analyses presented in this paper are freely available from our website. Moreover, our simulation and evaluation framework is made available as a web-sourceable script that consists of three modules: simulation, evaluation (running of the software packages) and metric computation. Each module can be extended, using simple wrapper functions to existing R-based code, ensuring that our comparison results are reproducible, extensible and relatively easy for the user to track exactly what code segments (and versions) were run.

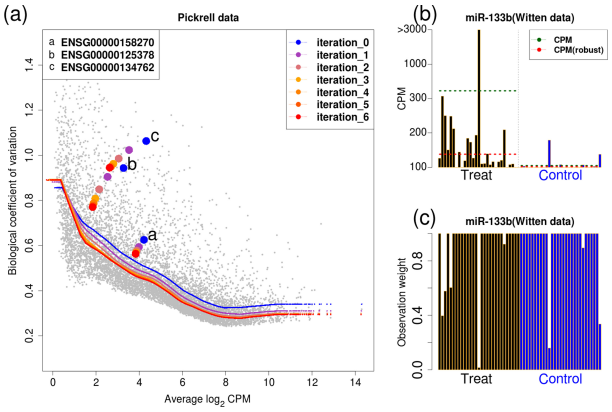

Figure 3. (**a**) For the random 5 versus 5 split of the Pickrell data (10) shown in Figure 1, the trajectories of overall trended dispersion and for the three individual genes are shown over six iterations of the edgeR-robust re-weighted estimation scheme. (**b**) A bar plot of miR-133b expression from Witten *et al.* (25), including an observation with very high count. (**c**) weights for miR-133b after six iterations of the re-estimation from edgeR-robust. Dashed lines in panel (b) shown the group-wise CPM before and after weighting.

In addition to R code, we make available a web-based shiny 'app' that can be used to look at simulation results across a wide number of conditions (34). Since there are often too many methods to be easily displayed together, our app gives users the ability to present results for a user-selected subset of methods; the results update automatically as the user selects different simulation settings.

Functional category analysis for outliers

To explore potential biological or technical factors that may manifest as outliers, we performed hypergeometric-based functional category analyses on the set of genes with weights less than some cutoff (here, set to 1) separately for each sample. Our goal with such an analysis is to identify possible biological or technical factors that affect a subset of genes for a particular experimental unit. In some cases, this may shed light on why the expression levels of some genes for a given sample are very different than that of their replicates. Furthermore, we can investigate whether the down-weighting is driven by technical factors. As a positive control for this, we compared the observed weights to the sample-specific guanine-cytosine (GC) effects observed in the Pickrell dataset (10,35).

RESULTS

edgeR-robust dampens the effect of outliers

To highlight how edgeR-robust dampens the influence of outliers, we return to the dataset shown in Figure 1. Figure 3a shows the trajectories for the three outliers in terms of their average log-CPM and dispersion estimates and how the dispersion-mean trend changes over six iterations of the edgeR-robust re-weighted estimation scheme. Although we have not studied convergence in depth, Supplementary Figure S3 highlights the change in parameter estimate by itera-

tion; most features 'converge' after a small number of iterations and we use a fixed number of iterations as a stopping rule. As expected, the outliers appear 'extreme' according to the model, as also reflected by their residuals. Extreme residuals are then down weighted, iteratively, and both the dispersion and average log-CPM estimates are updated (Figure 3a). In particular, we notice large changes to the regression (e.g. log-fold-change) and dispersion parameter estimates, which impose better accordance, in terms of dispersion–mean relationship, with the other features in the dataset. Notably, Figure 3a highlights a global drop in dispersion-mean trend after the iterative robust estimation, which suggests that outliers present in sufficient frequency may have a global effect on the statistical detection of DE within a dataset. Thus, we speculate that gains in statistical power (see sections below) may be achieved in part by this global drop in trended dispersion.

In their manuscript, Li and Tibshirani (27) show some extreme examples of outliers affecting differential count analysis of miRNA-seq data (in particular, see their Figure 2). Figure 3b shows one of those examples, mir-133b, and highlights the estimated mean CPM by group, before and after down-weighting; the observation weights after six iterations are shown in Figure 3c. Notably, for this example, there still exists strong evidence for differential expression, even after careful reassessment of the outlying observations.

Supplementary Table S1 gives the full details of these analyses, before and after re-weighting.

Simulation reflects real data

To test the method on a wide range of simulated settings, we first generate count data from a model that reflects real data as well as possible. As described in the 'Materials and Methods' section, we choose to take the joint distribution of estimated log-CPM and dispersion from a large dataset as the basis for the parameter settings and we use library sizes that mimic those from typical datasets. For example, the Pickrell dataset (10) consists of >50 replicates, which should represent a reasonably accurate reflection of the range of abundances observed, as well as, in particular, the dispersion–mean relationship. We generate all data from the NB model and introduce outliers by various mechanisms (see 'Materials and Methods' section). Supplementary Figure S4 shows the dispersion-mean trend for the Pickrell dataset (top left) and an example simulated dataset based on the estimated parameters (top right), respectively, as well as the marginal distributions of both log-CPMs and dispersion. The framework for these simulations (see 'Materials and Methods' section) is designed to take an initial dataset that seeds the simulation parameters, so datasets spanning the range of biological variation could easily be tested. Notably, we explored the Pickrell dataset for both the frequency of outliers (as detected by down-weighting; Supplementary Figure S5 gives cumulative distributions of weights) and the magnitude of the outliers relative to non-down-weighted observations (Supplementary Figure S6) to justify the use of simulation parameters. In particular, we note that the range of outlier deviations is within the range we use (e.g. multiplication factor between 1.5 and 10; Supplementary Figure S6). Meanwhile, samples from the Pickrell dataset exhibit outlier rates of 2–10% 'per sample' (depending on where a weight threshold is set), suggesting our choice of 10% (of features with a single outlier) is in fact a conservative amount of outliers that may be present.

Standard metrics across various methods for various simulation settings

Next, we present a representative simulation and performance results under a single 'reasonable' setting of the parameters. We sampled NB model parameters μ and ϕ from the joint distribution of estimates from the Pickrell data (10) (dataset); we filtered out the top 10% of the extreme dispersion values (analogous to filtering; see Supplementary Figure S1); 10 000 features were generated (nTags), with a 5 versus 5 two-group comparison (group); 10% of them are defined as DE genes (pDiff=.1), symmetrically (pUp=.5) with fold difference 3 (foldDiff=3); outliers are introduced to 10% of the features (pOutlier=.1) using the 'simple' outlier generation mechanism (outlierMech="S"); outliers are randomly distributed among all features; further details are described in the 'Materials and Methods' section. Original simulated counts and the counts with outliers introduced are separately recorded and all methods were run on both.

Figure 4 shows the set of standard metrics: panels (a)–(c) and (d)–(f) show false discovery plots, ROC curves and power numbers, respectively, for the original and original-with-outliers datasets under the setting of simulation parameters discussed above. Overall, the introduction of outliers results in more false positives (Figure 4a versus d) and/or less true positives at the same false positive rate (Figure 4b versus e). In the absence of outliers, all methods exhibit similar patterns of false discovery rates, with the Bayesian methods, ShrinkBayes and EBSeq having a slightly higher rate. Similarly, in terms of separating the truly DE from non-DE features using a *P*-value (or *P*-value-like score in the case of Bayesian methods), all methods are very close in performance. Furthermore, in the absence of outliers, edgeR, edgeR-robust and DESeq2 appear to have a slight edge in power at the method's 5% FDR, albeit the advantage is small (Figure 4c). When outliers are introduced, edgeR-robust shows some advantages over edgeR. In terms of statistical power, all methods drop in overall power with the introduction of outliers (Figure 4c versus f), while DESeq exhibits a spectacular drop. Notably, DESeq still maintains a good ranking of *P*-values (Figure 4f), but becomes very conservative due to the maximum-of-trend-and-individual dispersion policy; in this respect, presence of outliers affect the whole dataset (see Supplementary Figure S7).

Since the direction of differential expression and the outlier introduction are applied at random, we can further split the DE features according to the position of the outlier relative to the direction of change in abundance (Figure 4g–i); 'DEupOutlier' represents the situation where the outlier is added to the higher expressed group; 'DEdownOutlier' represents those features where the outlier was added to the lower expressed condition; 'DEnoOutlier' represents DE features with no introduced outlier). Notably, edgeR shows the highest power in the 'DEupOutlier' setting, but this is artificial since the introduction of the outliers ac-

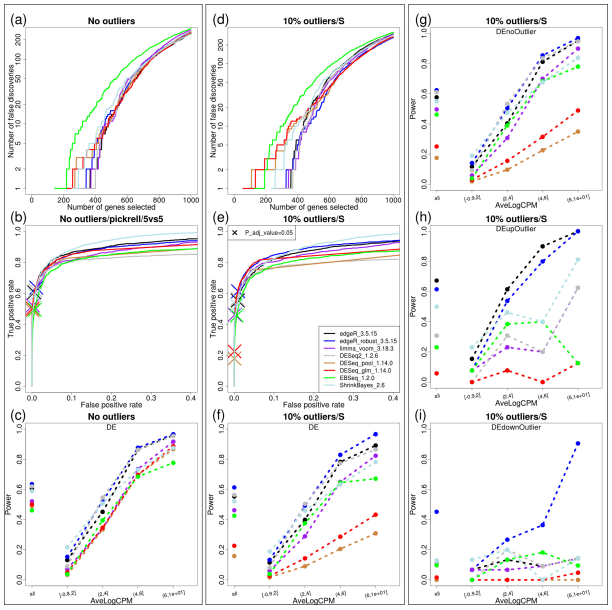

Figure 4. (**a**), (**b**) and (**c**) present FD, partial ROC (up to FP rate of 40%) and power plots (at each methods' 5% FDR) across several tested methods for datasets with no introduced outliers; (**d**), (**e**) and (**f**) show corresponding plots with datasets containing 10% outliers (i.e. 10% of genes have a single outlier) using 'S' method. (**g**), (**h**) and (**i**) split the results from panel (f) into three categories: features without outliers (g); outliers in the higher expression group (h); outliers in the lower expression group (i). All power results are shown as overall (single dot on the left of the plot) and split across five equally-sized average-log-CPM groups. The X on panels (b) and (e) highlights the achieved power (TP) according to each method's 5% FDR cutoff. Note that while panel (g) presents the situation with no outliers, there are outliers present in other features within the dataset and is therefore different from panel (c).

tually helps detection. The 'DEdownOutlier' is the situation where edgeR-robust comes to the forefront, as expected, given that outliers strongly eliminate the differential expression. In the absence of outliers, edgeR-robust still remains a strong competitor, closely followed by DESeq2, ShrinkBayes, limma-voom and edgeR.

It is also interesting, as a byproduct, to consider how well the methods identify outliers. In particular, we compared edgeR-robust's observation weights (using both Pearson and Deviance residuals) with DESeq2's Cook's distance metric (both at observation level and feature-wise maximum) to separate the simulated outliers. Supplementary Figure S8 shows an ROC curve depicting how well the observation weights (and other scores) separate outliers from non-outliers. Similarly, the default setting of DESeq2 leads to a similar tradeoff between false positives (here, falsely detected as an outlier) and false negatives (failing to identify an outlier) and Pearson residuals appear superior and are used for all further analyses with edgeR-robust. Notably, the edgeR-robust strategy smoothly identifies outliers and down-weights them according to the magnitude of discordance, instead of setting a hard threshold where statistical tests are no longer conducted. One byproduct of DESeq2's hard threshold is a loss of power (e.g. Figure 4 panels h and i), since genes with true differential expression as

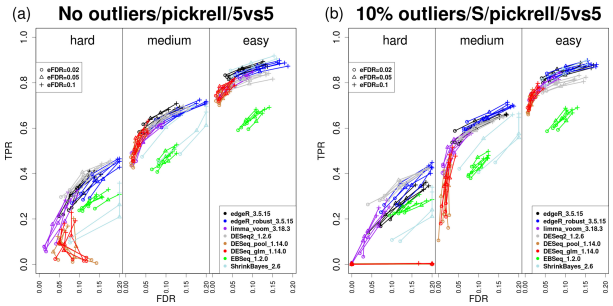

Figure 5. Power-to-achieved-FDR across hard (foldDiff $\in$ [2, 2.2]), medium (foldDiff $\in$ [3, 3.3]) and easy (foldDiff $\in$ [6, 6.6]) simulation settings. (**a**) No outliers; (**b**) 10% outliers. Y-axis shows TP rate and X-axis shows FD rate. Five simulations are shown for each method and each setting. Points are taken according to each method's FDR cutoffs at 0.02, 0.05 and 0.1.

well as outliers are excluded from statistical testing. We also tested DESeq2 after turning off the Cook's distance metric, which results in an expected sensitivity to outliers (Supplementary Figure S9). Although the focus has been on higher-in-magnitude outliers and indeed that is what we see more of (Supplementary Figure S6), lower outliers can be sufficiently detected and down-weighted (e.g. Supplementary Figure S10).

A shiny app to display pre-computed simulation results

The above discussion was in regard to a single dataset under a single set of simulation parameters. To provide a much wider scope of simulation settings, we created a web-based shiny app, that serves up pre-computed results over a range of simulation parameters, including different datasets, sample sizes and so on. In addition, it allows users to plot results for only the subset of desired methods and metrics from Figure 4. While new methods can only be added to the shiny app by us, existing simulations can be easily recreated in a local R environment or additional settings can be added, as described in the Supplementary Note. In general, the conclusions observed from the broader range of simulation settings (e.g. different magnitudes of DE, sample sizes) are in agreement with those mentioned above (see also Supplementary Figure S11).

Across multiple simulations over a range of settings, edgeR-robust is somewhat liberal but maintains a strong power-to-achieved-FDR tradeoff

To complement the simulation results for individual parameter settings, we endeavored to create a compact summary of a wider range of simulations and explore another important aspect of the comparison: do methods accurately control false discovery rate? Figure 5 shows a series of 15 simulations divided into three different blocks based on the degree of difficulty: 'hard' (foldDiff $\in$ [2, 2.2]), 'medium' (foldDiff $\in$ [3, 3.3]) and 'easy' (foldDiff $\in$ [6, 6.6]), including five simulations within each group to illustrate sampling variability. For each dataset, lines connect the true positive rates and achieved FDRs across three thresholds of the estimated

FDR (0.02, 0.05, 0.1). The rest of the simulation parameters are kept fixed: the NB model parameters originate from the Pickrell dataset (10), there are 10 000 features, we consider a two-group comparison (5 versus 5), 10% of features are DE and each dataset contains 10% 'S' outliers; comparisons for 3 versus 3 and 10 versus 10 are shown in Supplementary Figure S12.

Overall, there is a broad range of power-to-achieved-FDR tradeoffs and no method dominates. EBSeq appears to lack power and can be both liberal and conservative. In general, DESeq is conservative and achieves lower power, as reported earlier (6). Altogether, the collection of methods, such as limma-voom, edgeR, edgeR-robust and DESeq2 achieve similar power-to-achieved-FDR tradeoffs across the sample sizes, with perhaps a tendency to be more liberal in large sample sizes for edgeR and edgeR-robust. As expected and as highlighted above, edgeR-robust appears to have advantages in the presence of outliers, with only a minor decrease in power when no outliers are present. Thus, edgeR-robust achieves a good tradeoff between power at the same achieved FDR, even if the target FDR is not quite met. Notably, DESeq2 offers a small advantage in power at low log-fold-changes while suffering a little bit in power for higher log-fold-changes. In all cases, limma-voom controls FDR well and maintains high power.

Outliers may originate from technical or biological sources

While the strategy based on observation weights appears useful for dampening the effect of outliers in differential expression analysis, it may also be of interest to investigate the origin of such outlying observations. In some cases, we know of technical artefacts that affect the profile of RNA-seq expression data, such as sample-specific GC content biases, as highlighted and mitigated by the analyses of the HapMap consortium as well as in follow-up methodology development (e.g. conditional quantile normalization (35)). In this dataset, there are no experimental conditions to detect differential expression, so we fit an intercept-only model, using the iterative robust estimation scheme. Not surprisingly, we first observe that the two samples highlighted by Hansen *et al*. also exhibit a relatively higher number of down weighted observations (Figure 6a). As expected, the degree of down-weighting is strongly related to the GC content of the cDNA sequences of the genes involved (Figure 6b).

In an unrelated dataset from Blekhman (36) comparing expression in human male and female livers, we observe that the most significantly overrepresented functional categories were strongly associated with the set of down-weighted genes from a single sample (SRX014822and3, green circles in Figure 6c). These include several categories involving the extracellular matrix, as well as collagen catabolism and plasma membrane. We show the third most overrepresented category, 'extracellular matrix' (Figure 6c) because the size of this category allows individual genes to be visualized (further details are given in Supplementary Table S2). Although we cannot confirm the exact cause of the overrepresented gene ontology categories, we note that accumulation of collagen and excessive production of extracellular matrix proteins are associated with the development of liver fibrosis

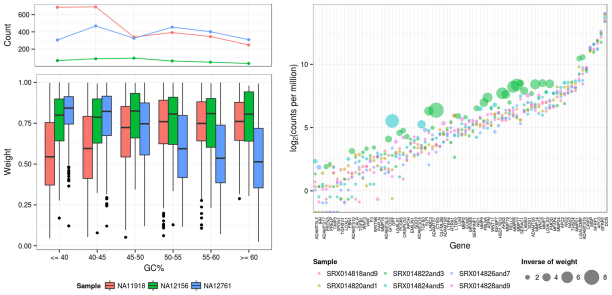

Figure 6. Technical ((**a**) and (**b**)) and biological (**c**) sources of outlier genes. The number of down weighted observations (a) and distribution of outlier weights as a function of the gene GC% in three samples from the HapMap RNA-Seq data (10) are plotted (b). Two of the samples shown (NA11918 and NA12761) were shown by Hansen *et al*. to have strong, opposing relationships between GC% and mapped reads per kilobase per million reads (RPKM). The third sample (NA12156) had the least number of genes down weighted after applying our robust down weighting procedure. (c) The log(CPM) and the inverse of the down weighting value for genes in the 'extracellular matrix' gene ontology category, where a value of one indicates no down weighting and larger inverse weights indicate stronger down weighting.

(e.g. 37,38), and we suggest that analyses such as these may assist biologists in identifying the source of outliers in gene expression.

DISCUSSION

Various method developers have shown that statistical methods for discerning differential expression from RNA-seq data represented as counts can be sensitive to outlying observations. In this report, we have studied in detail the effects of outliers on various approaches and developed a new method based on observation weights that can detect and dampen the effect of outliers. In fact, it requires a delicate tradeoff to maintain high power while at the same time achieving a decent resistance to the presence of outliers. In particular, it is difficult to know exactly what an outlier is and where the line should be drawn to identify it as such. In this respect, we take a 'smooth' approach of dampening their effects, when there is evidence to support departure from the model. We have also explored the origin of such outliers and in some cases, we may be able to identify either a technical or biological effect to explain them. Our robust approach follows the strategy of classical robustness methods that are commonly applied to the linear regression problem. In our approach, we adopted the calculation of the residuals and observation weights to the specifics of the flexible dispersion estimation and standard GLM regression estimation of the negative binomial model.

As mentioned above, one reason that edgeR is sensitive to outlying observations is that the dispersion estimate used in the downstream inference is pulled toward the dispersion-mean trend, which may underestimate the variability. Therefore, another way to dampen the effect of outliers is to decrease the degree of moderation toward the dispersion-mean trend. Although we have not studied it here, there is again a delicate tradeoff between the degree of moderation to use and the average inference performance;

it still remains an open question as to how exactly to set this value for a given dataset.

Though motivated and tested on real datasets, we employed simulations to explore the broad range of possible settings and developed a comprehensive system for such evaluations. Our strategy to mimick real datasets is to take the joint distribution of mean and dispersion estimates from a large dataset as the basis for parameters to sample from. From such a dataset, outliers and differential expression at a specified level can be readily introduced. In fact, because these are estimates and not true values, we expect the sampled dispersion to potentially exhibit more variation than observed in a real dataset. In terms of evaluating the different methods across the spectrum of simulation settings, it is important to consider it from all points of view: false discoveries amongst the list of top called features, the ability to separate the truly differential from non-differential (i.e. ranking by statistical evidence), the statistical power at thresholds that are typically used in practice and the degree to which methods achieve their purported false discovery rates.

Overall, the observation weight robust method performs well and achieves the goal of suffering only minimal loss of power, while maintaining resistance to introduced outliers. We have investigated the outlier policy in other packages and highlight that smoothly down-weighting outlying observations appear preferable. In DESeq, a hard line against outliers is taken by using the maximum of a dispersion-mean trend and the individual estimate; with the addition of outliers, this has a global effect of increasing the variance to all features and gives a resulting loss of power. In DESeq2, a Cook's distance metric is used to remove features with outliers entirely from further consideration; in this case, features that have outliers and differential expression are excluded, with a potential loss of power. It is somewhat of a philosophical decision as to whether to completely filter out features or to down-weight them; the observation weight strategy allows both.

Another important consideration is the required sample size to be able to achieve estimators that are resistant to outliers. Indeed, the lowest levels of replication (e.g. two samples per condition) will not be sufficient. The minimum level of replication to dampen effects of outliers is three samples per condition, but this is the limit of any robust procedure.

With the simulation system that we have created, we can now make a call to the community of both developers and users to check the effect of various settings. All that is required to test a new method and compare it against existing methods is to write a wrapper function with the correct inputs and outputs. In addition, if the exact simulation settings that we use in this report are not adequate, we can easily extend this framework into an open testing system that allows additional variations on the sampling model, perhaps including additional distributions or constructed truths, such as plasmodes (39).

The current edgeR framework does not always achieve its false discovery rate target. However, even if it is forced to be more conservative, it still achieves power as good or better than existing approaches across the simulation settings that we have tested, even with the addition of observation weights. The exact source of the liberality is beyond the scope of the current investigation, but there may be room for improvement, such as borrowing ideas from small sample asymptotic approximations (40).

CONCLUSION

We developed an approach to dampen the effect of outliers on count-based differential expression analyses. Overall, the method appears to achieve the desired 'efficiency': a resistance to outliers while maintaining high power. We provided an implementation for the edgeR Bioconductor package, but the re-weighting idea could easily be adopted to other packages. In addition, we developed an extensible simulation system (at the count table level) that readily performs the simulations based on an existing dataset and provides the infrastructure for producing the standard battery of evaluations. In particular, this allows new methods or variations (e.g. alternative settings) of existing packages to be quickly explored. Instead of preparing a large number of Supplementary Figures, we provide an interactive web-based shiny 'app' to display simulation results across a broad range of simulation settings.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online, including [1,2].

ACKNOWLEDGMENTS

The authors wish to thank all members of the Robinson laboratory for helpful discussions and in particular, Olga Nikolayeva, Gosia Nowicka, Katarina Matthes and Charity Law for careful reading of an earlier version of the manuscript; we also thank members of the Baudis and von Mering groups for useful feedback. We thank Gordon Smyth and Aaron Lun for aspects of the edgeR implementation.

FUNDING

SNSF Project Grant [143883]; European Commission through the 7th Framework Collaborative Project RADIANT [305626]. Funding for open access charge: SNSF Project Grant [143883]; European Commission through the 7th Framework Collaborative Project RADIANT [305626].
Conflict of interest statement. None declared.

REFERENCES

1. Wang,Z., Gerstein,M. and Snyder,M. (2009) RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.*, **10**, 57–63.
2. Anders,S., McCarthy,D.J., Chen,Y., Okoniewski,M., Smyth,G.K., Huber,W. and Robinson,M.D. (2013) Count-based differential expression analysis of RNA sequencing data using R and Bioconductor. *Nat. Protocols*, **8**, 1765–1786.
3. Li,B. and Dewey,C.N. (2011) RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics*, **12**, 323.
4. Grabherr,M.G., Haas,B.J., Yassour,M., Levin,J.Z., Thompson,D.A., Amit,I., Adiconis,X., Fan,L., Raychowdhury,R., Zeng,Q. *et al.* (2011) Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nat. Biotechnol.*, **29**, 644–652.
5. Frazee,A.C., Langmead,B. and Leek,J.T. (2011) ReCount: a multi-experiment resource of analysis-ready RNA-seq gene count datasets. *BMC Bioinformatics*, **12**, 449.

6. Soneson,C. and Delorenzi,M. (2013) A comparison of methods for differential expression analysis of RNA-seq data. *BMC Bioinformatics*, **14**, 91.
7. Rapaport,F., Khanin,R., Liang,Y., Pirun,M., Krek,A., Zumbo,P., Mason,C.E., Socci,N.D. and Betel,D. (2013) Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data. *Genome Biol.*, **14**, R95.
8. McCarthy,D.J., Chen,Y. and Smyth,G.K. (2012) Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Res.*, **40**, 1–10.
9. Anders,S. and Huber,W. (2010) Differential expression analysis for sequence count data. *Genome Biol.*, **11**, R106.
10. Montgomery,S.B., Sammeth,M., Gutierrez-Arcelus,M., Lach,R.P., Ingle,C., Nisbett,J., Guigo,R. and Dermitzakis,E.T. (2010) Transcriptome genetics using second generation sequencing in a Caucasian population. *Nature*, **464**, 773–777.
11. Law,C.W., Chen,Y., Shi,W. and Smyth,G.K. (2014) Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biol.*, **15**, R29.
12. Gentleman,R.C., Carey,V.J., Bates,D.M., Bolstad,B., Dettling,M., Dudoit,S., Ellis,B., Gautier,L., Ge,Y., Gentry,J. *et al.* (2004) Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol.*, **5**, R80.
13. Robinson,M.D., McCarthy,D.J. and Smyth,G.K. (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139–140.
14. Van De Wiel,M.A., Leday,G. G.R., Pardo,L., Rue,H.v., Van Der Vaart,A.W. and Van Wieringen,W.N. (2013) Bayesian analysis of RNA sequencing data by estimating multiple shrinkage priors. *Biostatistics*, **14**, 113–128.
15. Hardcastle,T.J. and Kelly,K.A. (2010) baySeq: empirical Bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, **11**, 422.
16. Wu,H., Wang,C. and Wu,Z. (2013) A new shrinkage estimator for dispersion improves differential expression detection in RNA-seq data. *Biostatistics*, **14**, 232–243.
17. Carroll,R.J. and Pederson,S. (1993) On robustness in the logistic regression model. *J. R. Stat. Soc. Ser. B*, **84**, 693–706.
18. Cantoni,E. and Ronchetti,E. (2001) Robust inference for generalized linear models. *J. Am. Stat. Assoc.*, **96**, 1022–1030.
19. Alqallaf,F. and Agostinelli,C. (2013) Robust inference in generalized linear models, in press.
20. Agostinelli,C. (2002) Robust model selection in regression via weighted likelihood methodology. *Stat. Probab. Lett.*, **56**, 289–300.
21. Cox,D.R. and Reid,N. (1987) Parameter orthogonality and approximate conditional inference. *J. R. Stat. Soc. Ser. B Methodol.*, **49**, 1–39.
22. Robinson,M.D. and Smyth,G.K. (2008) Small-sample estimation of negative binomial dispersion with applications to SAGE data. *Biostatistics*, **9**, 321–332.
23. Robinson,M.D. and Smyth,G.K. (2007) Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, **23**, 2881–2887.
24. Fox,J. (2002) Robust Regression. *Behav. Res. Methods*, **1**, 1–8.
25. Witten,D., Tibshirani,R., Gu,S.G., Fire,A. and Lui,W.-O. (2010) Ultra-high throughput sequencing-based small RNA discovery and discrete statistical biomarker analysis in a collection of cervical tumours and matched controls. *BMC Biol.*, **8**, 58.
26. Cheung,V.G., Nayak,R.R., Wang,I.X., Elwyn,S., Cousins,S.M., Morley,M. and Spielman,R.S. (2010) Polymorphic Cis- and Trans-Regulation of Human Gene Expression. *PLoS Biol.*, **8**, 14.
27. Li,J. and Tibshirani,R. (2013) Finding consistent patterns: a nonparametric approach for identifying differential expression in RNA-Seq data. *Stat. Methods Med. Res.*, **22**, 519–39.
28. Leng,N., Dawson,J.A., Thomson,J.A., Ruotti,V., Rissman,A.I., Smits,B. M.G., Haag,J.D., Gould,M.N., Stewart,R.M. and Kendziorski,C. (2013) EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics*, **29**, 1035–1043.
29. Robinson,M.D. and Oshlack,A. (2010) A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biol.*, **11**, R25.
30. Rue,H.v., Martino,S. and Chopin,N. (2009) Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *J. R. Stat. Soc. Ser. B: Stat. Methodol.*, **71**, 319–392.
31. Martins,T.G., Simpson,D., Lindgren,F. and Rue,H. (2012) Bayesian computing with INLA: new features, in press .
32. Efron,B., Tibshirani,R., Storey,J.D. and Tusher,V. (2001) Empirical Bayes analysis of a microarray experiment. *J. Am. Stat. Assoc.*, **96**, 1151–1160.
33. Benjamini,Y. and Hochberg,Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B: Methodol.*, **57**, 289–300.
34. RStudio shiny: Web Application Framework for R (Version 0.7.0) (2013).
35. Hansen,K.D., Irizarry,R.A. and Wu,Z. (2012) Removing technical variability in RNA-seq data using conditional quantile normalization. *Biostatistics*, **13**, 204–216.
36. Blekhman,R., Marioni,J.C., Zumbo,P., Stephens,M. and Gilad,Y. (2010) Sex-specific and lineage-specific alternative splicing in primates. *Genome Res.*, **20**, 180–189.
37. Herbst,H., Wege,T., Milani,S., Pellegrini,G., Orzechowski,H., Bechstein,W., Neuhaus,P. and Schuppan,D. (1997) Tissue inhibitor of metalloproteinase-1 and -2 RNA expression in rat and human liver fibrosis. *Am. Soc. Invest. Pathol.*, **150**, 51647–51659.
38. Asselah,T., Bièche,I., Laurendeau,I., Paradis,V., Vidaud,D., Degott,C., Martinot,M., Bedossa,P., Valla,D., Vidaud,M. *et al.* (2005) Liver gene expression signature of mild fibrosis in patients with chronic hepatitis C. *Gastroenterology*, **129**, 2064–2075.
39. Reeb,P. and Steibel,J. (2013) Evaluating statistical analysis models for RNA sequencing experiments. *Front. Genet.*, **4**, 178 .
40. Di,Y., Sarah,C.E., Daniel,W.S., Kimbrel,J.A. and Chang,J.H. (2013) Higher order asymptotics for negative binomial regression inferences from RNA-sequencing data.*Stat. App. Genet. Mol. Biol.*, **12** , 49 –70 .

CHAPTER II

benchmarkR: an R package for benchmarking genome-scale methods

Xiaobei Zhou, Charity W Law and Mark D. Robinson

Paper published in *bioRxiv (2015)*

benchmarkR: an R package for benchmarking genome-scale methods

Xiaobei Zhou[1,2], Charity W. Law[1,2], and Mark D. Robinson [1, 2, *]

[1]Institute of Molecular Life Sciences, University of Zurich, CH-8057 Zurich, Switzerland
[2]SIB Swiss Institute of Bioinformatics, University of Zurich, CH-8057 Zurich, Switzerland

April 16, 2015

[*]To whom correspondence should be addressed. Tel: +41 44 635 48 48; Fax: +41 44 635 68 68; Email: mark.robinson@imls.uzh.ch

Abstract

1 Summary:

benchmarkR is an R package designed to assess and visualize the performance of statistical methods for datasets that have an independent truth (e.g., simulations or datasets with large-scale validation), in particular for methods that claim to control false discovery rates (FDR). We augment some of the standard performance plots (e.g., receiver operating characteristic, or ROC, curves) with information about how well the methods are calibrated (i.e., whether they achieve their expected FDR control). For example, performance plots are extended with a point to highlight the power or FDR at a user-set threshold (e.g., at a method's *estimated* 5% FDR). The package contains general containers to store simulation results (`SimResults`) and methods to create graphical summaries, such as receiver operating characteristic curves (`rocX`), false discovery plots (`fdX`) and power-to-achieved FDR plots (`powerFDR`); each plot is augmented with some form of calibration information. We find these plots to be an improved way to interpret relative performance of statistical methods for genomic datasets where many hypothesis tests are performed. The strategies, however, are general and will find applications in other domains.

2 Availability:

The **benchmarkR** package is available from:
`https://github.com/markrobinsonuzh/benchmarkR`

3 Contact:

mark.robinson@imls.uzh.ch

4 Introduction

The burden of proof in developing new statistical methods for inferring differences (e.g., changes in abundance) in genomic datasets is improved performance against existing methods. Methodologists typically resort to simulations since there is limited availability of large-scale validation datasets. To evaluate simulation performance (or performance with sufficient validation information), various metrics and plots are typically used, including but not limited to receiver operating characteristics (ROC) curves, which shows the tradeoff between true positive rates (TPR, or sensitivity, or power) and false positive rate across many cutoffs [1, 2], or false discovery (FD) plots, which highlight the cumulative number of false discoveries amongst the top ranked features.

While a method's ability to give a good ranking is important, statistical methods typically build in some kind of adjustment to control the rate of errors made; in genomics, this typically takes the form of false discovery rate (FDR) control. Therefore, in these settings, it is of interest not only to know about detection performance (i.e., how well a statistical method separates true changes from false), but if and how well the error is controlled. To allow ourselves and the community more flexible ways to visualize additional information with respect to "calibration" in standard plots, such as ROC and FD plots, we developed the R-based **benchmarkR** package. In particular, we promote the use of a new variation: *power-to-achieved-FDR* plots at a small number of typical thresholds to directly contrast detection performance and error control. We find this plot to be the simplest way to digest both angles.

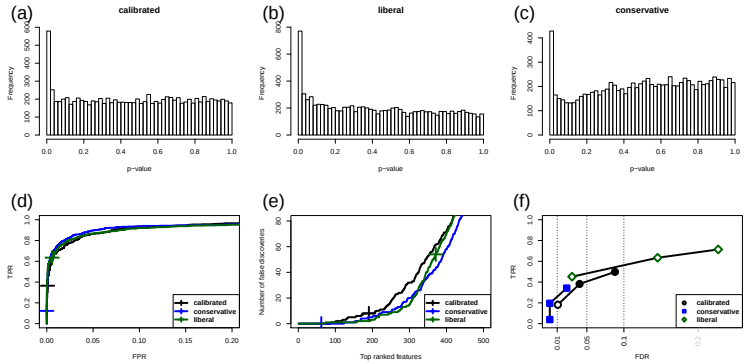

Figure 1: A hypothetical example of a calibrated, liberal and conservative statistical method in genomics. Panels (a)-(c) should P-value distributions. Panels (d), (e), (f) show an ROC curve (`rocX`), a false discovery plot (`fdX`) and a power-versus-achieved-FDR plot (`powerFDR`), respectively. The code to regenerate this plot is available as Supplementary Material.

Figure 1 gives a simple but illustrative example. Suppose there is a simulation with a total of 10,000 features, of which 1,000 are truly differential. In this toy example, all features were generated for 3 replicates versus 3 replicates from a normal distribution with mean 0 and variance 1, except for the 1,000 differential features, which had a shifted mean (R code is available as

Supplementary Material). If a method happens to systematically under- or overestimate the variance, this will lead to some initial clues in the distribution of raw P-values, but it may not compromise the method's ability to rank differential features. A *calibrated* method should show a mixture of uniform P-values with the differential features showing strong statistical evidence as a peak at the low end (Figure 1a). A (systematically) *liberal* method will tend to overstate the statistical evidence and push all P-values toward 0 (Figure 1b), whereas a *conservative* method will push P-values towards 1 (Figure 1c), relative to a calibrated method (in our toy example, this calibration is modified through the variance estimates). In practice, P-value distributions may be hard to diagnose since a combination of factors will affect their overall shape and other problems may arise, such as correlation of observations, model misspecification or outliers. Importantly, calibration is not well represented in an ROC curve (Figure 1d). Despite the differences in statistical calibration that we have introduced to the toy example, the ROC curve cannot relay any difference in performance (in the toy example, the calibration does not strongly affect the ranking). In addition, ROC curves can actually be misleading because it is not known where a particular usage of a method (e.g., FDR=5%) will lie on the curve. For example, a method could have a great ability to rank features (a high ROC curve and area under the curve), but it may be extremely conservative and thus not very useful in practice. The ROC curve plotted using the **benchmarkR** package (Figure 1d; `rocX` method) highlights the point on each ROC curve that corresponds to the method's estimated 5% FDR threshold; however, it is important to note that the method does not necessarily achieve this level of control. An alternative method to look at simulation results is an FD plot (Figure 1e), where the cumulative number of false discoveries is displayed amongst the top ranked differential features; fewer FDs are desirable. The **benchmarkR** package provides a variation of this plot (using the `fdX` method) that adds the location of the method's operating position (e.g., FDR=5%). This allows the methodologist to get a sense of whether methods are adequately controlling their FDR. Pushing this further, we find a power-to-achieved-FDR plot (via the `powerFDR` method) to be a concise summary of both angles (Figure 1f). In this plot, several typical FDR thresholds are used (e.g., FDR=1%, 5%, 10%) and for each threshold, the method's performance in terms of power and achieved FDR are plotted, with a line joining the different cutoffs. For these plots, it is desirable when the method is able to control the FDR, which would require the method's X-axis point to be on the left side of the corresponding threshold line; if this occurs, the default plotting system in **benchmarkR** will use a filled-in symbol whereas if the error is not controlled, an open symbol will be used.

5 Implementation

A typical use of the **benchmarkR** package may look like the following:

```
library("benchmarkR")

# create container for results
re <- SimResults(pval,labels)

# 3-panel plot
```

```
benchmarkR(re)

# individual plots
rocX(re)
fdX(re)
powerFDR(re)
```

where `pval` (a vector or matrix) and `labels` (a vector) give the scores and labels, respectively. The `benchmarkR` function is simply a wrapper that makes a 3-panel plot consisting of `rocX`, `fdX` and `powerFDR`. Each individual plot is highly customizable; see the package vignette for further details.

Acknowledgement

The authors wish to thank all members of the Robinson laboratory for helpful discussions. We wish to acknowledge funding from an SNSF Project Grant (143883) and the European Commission through the 7th Framework Collaborative Project RADIANT (Grant Agreement Number: 305626).

Conflict of Interest: none declared.

References

[1] Tobias Sing, Oliver Sander, Niko Beerenwinkel, and Thomas Lengauer. ROCR: Visualizing classifier performance in R. *Bioinformatics*, 21(20):3940–3941, 2005.

[2] John A Swets. Signal detection theory and ROC analysis in psychology and diagnostics : collected papers. In *Scientific psychology series*, page Chp 11. Psychology Press (12 Jun. 1996), 1996.

CHAPTER III

Do count-based differential expression methods perform poorly when genes are expressed in only one condition?

Xiaobei Zhou and Mark D. Robinson

Paper published in *Genome Biology (2015), 16, 222*

Zhou and Robinson *Genome Biology* (2015) 16:222
DOI 10.1186/s13059-015-0781-3

Genome Biology

CORRESPONDENCE Open Access

Do count-based differential expression methods perform poorly when genes are expressed in only one condition?

Xiaobei Zhou[1,2] and Mark D. Robinson[1,2*]

Abstract

A response to 'Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data' by Rapaport F, Khanin R, Liang Y, Pirun M, Krek A, Zumbo P, Mason CE, Socci ND and Betel D in Genome Biology, 2013, 14:R95

Background

Statistical methods for determining transcriptional changes between (replicated) groups of cell populations using RNA sequencing (RNA-seq) data are now quite mature. Several themes that emerged from the past decade of modeling microarray data apply analogously to RNA-seq data: parameter moderation is critical, multiple testing corrections are necessary and flexible frameworks (e.g., linear models) to account for the effect of covariates are essential. For RNA-seq data, popular packages such as edgeR, DESeq and DESeq2 [1–3] perform detailed modeling of the dispersion–mean relationship, with variations on fitting a dispersion by mean trend and moderating estimates toward the trend. Likewise, careful modeling of the mean–variance relationship of transformed data has been proven effective, essentially 'unlocking' the world of heteroskedastic linear regression [4].

A recent report in Genome Biology from Rapaport and co-authors claimed that some methods, namely PoissonSeq [5] and limma [6], 'have improved modeling of genes expressed in one condition', where they showed a striking difference in the ability to separate differential expression (DE) [7]. From a methodological perspective, this result caught our interest and prompted us to understand how aspects of the all-zero-in-one-condition manifest undesirable properties in count-based models. Briefly, (i) we found a coding error in the calculation of edgeR's signal-to-noise (S/N) metric and (ii) our re-analysis suggests that count-based methods perform as well or better than other methods, counter to the original conclusion.

The Rapaport manuscript is an excellent model of modern bioinformatics research, in terms of making processed data and code available that reproduce figures from their manuscript. In many cases, the small details can be important and this open-source model facilitates quick access in understanding precisely what settings were used. We fully support this model and by default, also make our code available. In this correspondence, we investigate the genesis of differences in method performance that Rapaport and co-authors observed and provide our view of how performance results can be sensitive to decisions made.

Genes expressed in only one condition

We first briefly summarize the analysis that Rapaport and colleagues reported, with respect to the all-zero-in-one-condition case.

Using gene-level read counts, they isolated genes that exhibit zero-counts across all replicates of a single condition; in general, the number of such genes is related to the depth of sequencing dedicated to each sample, with deeper sequencing resulting in fewer such cases. The dataset in question, comparing GM12892 cells to H1-hESC cells [8], with three and four replicates, respectively, had typical read depths for such experiments (16–39 million mapped reads). They used the following pipeline: (i) from the count table, generate DE *P* values for several

*Correspondence: mark.robinson@imls.uzh.ch
[1] SIB Swiss Institute of Bioinformatics, University of Zurich, CH-8057 Zurich, Switzerland
[2] Institute of Molecular Life Sciences, University of Zurich, CH-8057 Zurich, Switzerland

methods; (ii) calculate S/N using 'normalized' data; (iii) plot negative log *P* value versus S/N, where they expect a monotonic positive dependency (correlation); and, (iv) generate receiver operating characteristic (ROC) curves with thresholds on the S/N to illustrate the ability to separate low S/N (<3) from high S/N (>3).

They highlighted that count-based methods such as DESeq and edgeR, which infer changes in expression via the negative binomial (NB) model, do not perform very well in this case. It is worth noting that this is a non-standard use of ROC curves: here, all genes are strictly DE, but they vary in their magnitude of change. So, the ROC curve represents the ability to separate low S/N from high S/N. Rapaport and colleagues postulated that the NB model reduces to Poisson (dispersion ≈ 0) and lacks the ability to handle the 'wide variations' in gene counts among replicate libraries. Our aim with this report is to understand the origins of this result, whether it is a shortcoming of the dispersion estimation strategy or in the inference machinery, since parameter estimates are on the boundary of the parameter space.

Signal-to-noise has some potential limitations

We became interested in the suitability and robustness of the S/N metric itself, since it forms the basis for the 'truth' in Rapaport's ROC result. In theory, the S/N of the non-zero observations should accurately reflect the significance of model-based *P* values for the expressed-in-one-condition versus zero differences. In practice, however, there are some potential difficulties: the sample sizes are small and therefore, the S/N itself is subject to considerable estimation uncertainty; it is well known that for count data the variance is intimately tied to the mean, so it is not clear whether S/N should be calculated on a linear scale. In addition, a notable aspect of the Rapaport ROC comparison is that while the same S/N cutoff (= 3) is used across all methods, different sets of true and false DE labels are used; this makes the curves difficult to compare, since both the truth and score change by method. We explore these issues here.

Table 1 and Fig. 1 give illustrative examples of the differences in the originally calculated S/N between edgeR and voom. Figure 1 gives a scatter plot of S/N calculated on each method's normalized data, highlighting in some cases large differences. Table 1 shows the top ten genes for both edgeR's (estimated) false discovery rate (FDR) and calculated S/N. (The full table of zero-counts, differential statistics and S/N is given in Additional file 1.) Here, it is evident that several genes that show little evidence for DE, have very high S/N for edgeR but not for voom (e.g., C17orf66, TM4SF19 and NPY1R). However, the *P* values seem to reflect appropriately the magnitude of evidence for DE, although they are on drastically different scales between edgeR and voom (see 'Discussion' for further commentary on this). In addition, several genes that show the largest evidence against the null hypothesis (e.g., PLEK, MS4A1, etc.) show relatively low S/N for edgeR and would be counted as false discoveries (according to a S/N = 3 cutoff), while voom's higher S/N would result in these counted as true positives. Therefore, it is not clear whether the ROC curve reflects the accuracy of the S/N calculation itself or of the statistical method's capabilities. Upon investigation, the differences in S/N exhibited in Fig. 1 resulted from a code error in the original report (see Additional file 2: Fig. S1).

Another aspect to understand is the scale on which the S/N is calculated. As is well known with count data, the variance is related to the mean. In particular, using the NB parameterization with mean μ and variance $\mu(1 + \mu\phi)$, the theoretical S/N is then:

$$\mathrm{S/N} = \frac{1}{\sqrt{1/\mu + \phi}},$$

which implies $\mathrm{S/N} \rightarrow \phi^{-1/2}$ with sufficiently large μ. Thus, depending on the mean, the S/N calculation is capturing the (inverse square root of) dispersion. For the ENCODE data, this relationship is shown in Additional file 2: Fig. S2. Since the S/N calculations are most relevant when the variance is independent of the mean, we explored how transforming the data, which alters the mean–variance relationship, affects the results of the ROC comparisons that Rapaport and co-authors performed. Figure 2a–c show mean–variance relationships for S/N calculated on different scales and Fig. 2d–f highlight their corresponding ROC performances. In all cases, the true/false labels for the ROC curves are the same across methods; specifically, counts-per-million from edgeR are used to base the S/N calculation. Since the scale of data changes the scale of S/N, true genes are selected according to S/N > 40th percentile and false as the lowest 20 % of S/N to give a gray zone of uncertainty in the middle. (Additional file 2: Fig. S3 gives alternative settings for these cutoffs, but the results are unaffected.) Figure 2d shows similar results to the original Rapaport study, whereas Fig. 2e, f show a remarkable reversal in performance, giving clear evidence for our earlier concern regarding the S/N calculation.

Count-based methods perform well on zero-in-one-condition simulation

Given recent efforts in simulating RNA-seq count tables [9–11], we tried to create a representative simulation for the zero-in-one-condition situation. The simulation was designed as follows: (i) generate a dataset with no DE and (ii) randomly select genes across the spectrum of expression levels and set counts for one condition (chosen at random) equal to zero to represent 'true' DE genes.

Table 1 Top ten genes originally calculated using S/N (for edgeR-normalized data; first ten rows) and top ten genes calculated using FDR for DE (edgeR *P* values; second ten rows). The table includes the counts-per-million table (A = GM12892 and B = H1-hESC), S/N and estimated false discovery rate (FDR) for edgeR and limma-voom for the ENCODE dataset comparing three replicates of GM12892 to four replicates of H1-hESC

Id	A1	A2	A3	B1	B2	B3	B4	edgeR S/N	edgeR FDR	voom S/N	voom FDR
MIPOL1	0.0	0.0	0.0	237.1	232.5	226.0	227.5	45.75	7.47e−31	3.79	3.32e−07
AQP4	0.0	0.0	0.0	46.1	45.0	46.7	44.4	43.87	1.44e−14	3.90	1.99e−06
FAM19A4	0.0	0.0	0.0	142.1	131.1	143.8	131.4	20.21	1.91e−24	4.06	4.72e−07
C17orf66	2.3	2.1	2.1	0.0	0.0	0.0	0.0	16.96	1.89e−02	2.72	7.40e−04
TM4SF19	3.5	3.2	3.2	0.0	0.0	0.0	0.0	16.96	3.00e−03	2.72	2.39e−04
SOX1	0.0	0.0	0.0	7.5	6.7	6.5	6.3	13.02	8.37e−05	4.33	1.27e−04
HPGD	0.0	0.0	0.0	22.6	21.0	19.6	19.0	12.97	5.08e−09	4.56	7.48e−06
LOC100131176	0.0	0.0	0.0	17.9	15.3	18.7	17.2	12.02	5.14e−08	3.51	1.49e−05
ZNF385D	0.0	0.0	0.0	135.5	155.0	155.0	132.3	11.80	2.60e−25	4.67	3.70e−07
NPY1R	0.0	0.0	0.0	209.8	179.9	179.3	208.5	11.38	1.33e−27	2.95	6.77e−07
PLEK	25 082.8	12 622.5	11 394.8	0.0	0.0	0.0	0.0	2.16	1.79e−216	16.11	9.36e−09
MS4A1	25 455.1	14 937.7	12 886.8	0.0	0.0	0.0	0.0	2.63	2.62e−215	7.26	1.60e−08
SLAMF1	7 407.2	4 859.3	4 283.2	0.0	0.0	0.0	0.0	3.32	2.98e−165	5.11	2.53e−08
CCL3	11 057.5	3 413.1	3 544.1	0.0	0.0	0.0	0.0	1.37	4.62e−165	5.13	2.15e−08
FCRLA	7 742.0	2 979.1	3 879.8	0.0	0.0	0.0	0.0	1.92	1.01e−161	6.08	1.84e−08
RGS1	9 939.5	9 967.3	7 741.6	0.0	0.0	0.0	0.0	7.22	4.53e−159	2.66	1.44e−07
DPPA4	0.0	0.0	0.0	14 580.2	15 215.1	14 745.3	10 617.3	6.47	2.37e−158	12.02	1.84e−08
TDGF1	0.0	0.0	0.0	15 699.8	15 481.1	13 374.5	8 522.5	3.98	6.37e−157	15.48	1.84e−08
SFRP2	0.0	0.0	0.0	14 673.3	15 229.5	13 067.2	7 234.4	3.43	1.84e−153	9.87	2.15e−08
BLK	9 943.0	2 954.7	2 351.8	0.0	0.0	0.0	0.0	1.20	2.98e−147	3.28	5.17e−08

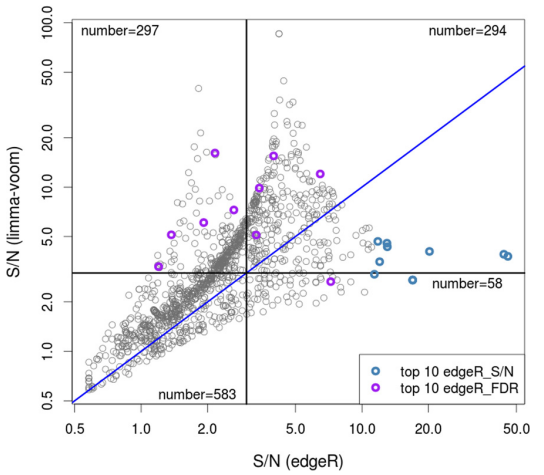

Fig. 1 Scatter plot of S/N (signal-to-noise) for limma-voom and edgeR for the ENCODE dataset. S/N is calculated from the non-all-zero condition. More information about the colored points is given in Table 1

As previously, we sampled NB mean and dispersion estimates from the joint distribution of estimates using a large dataset (here, from [12]) and filtered out extreme dispersion values. Altogether, 30,000 features were generated in a 5 versus 5 two-group comparison and zero-counts were introduced to 5 % of the features. To reflect that zeros occur somewhat more often at lower expression across various datasets (see Additional file 2: Fig. S4), we increased the frequency of zero-counts at low expression strength.

Based on the results of this simulation (Fig. 3), ROC curves with the method's 5 % FDR highlighted (panel a) and plots of true positive rate versus achieved FDR (panel b), we again see that count-based models perform well in the zero-in-one-condition situation. In addition, we explored the postulation that the NB model is reduced to a Poisson in these zero-count situations. By comparing the dispersion estimates calculated from the single non-zero condition to the original non-zero-in-both-conditions data, it does not appear that the dispersion estimates are drastically reduced (see Additional file 2: Fig. S5).

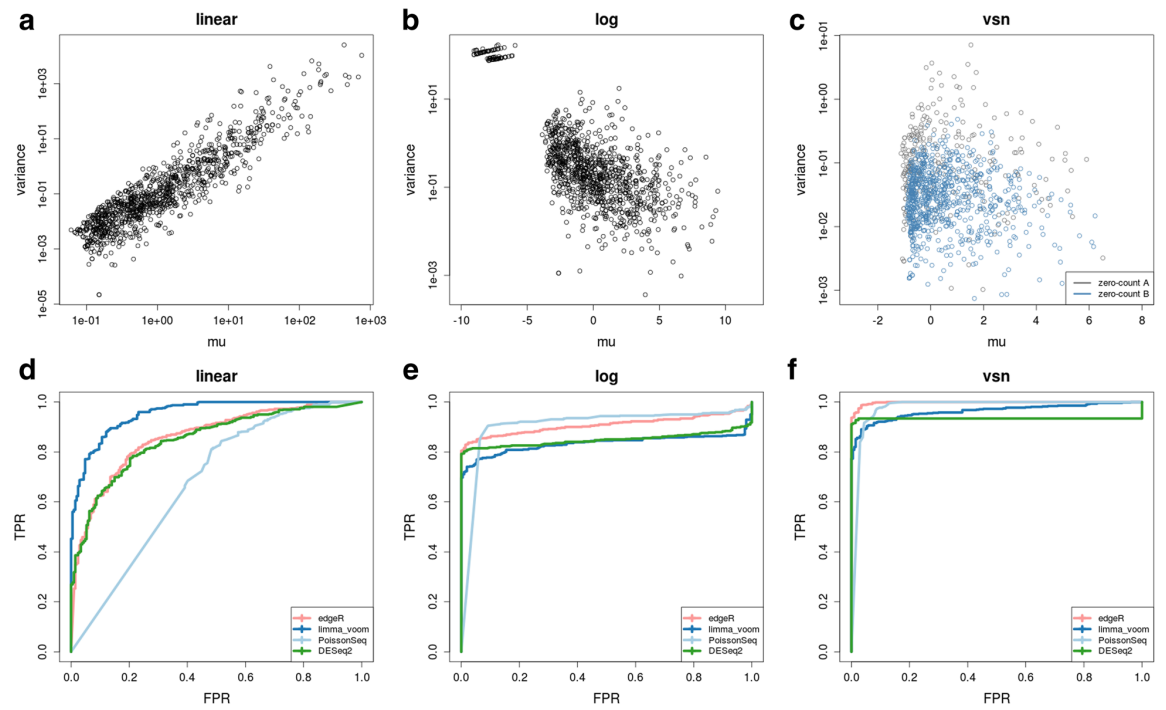

Fig. 2 The effect of scale that signal-to-noise is calculated on. **a**–**c** Mean–variance relationships for different scales of the original all-zero-in-one-condition data. **d**–**f** Corresponding ROC curves for the ENCODE dataset (GM12892 cells to H1-hESC), using S/N to set the true labels. Here, the signal-to-noise (S/N) is calculated from (trimmed mean of M-values-normalized) counts-per-million and used for all methods. Linear is equivalent to Rapaport's method, where S/N is calculated on the counts-per-millions. Log represents S/N calculated on log-transformed counts-per-million. vsn represents S/N calculated on variance-stabilized data [14]. ROC curves employ the same labels across all methods: the top 40 % of S/N are used as true DE genes whereas the lowest 20 % are false. Each method's *P* value is used for ranking the genes. *FPR* false positive rate, *TPR* true positive rate

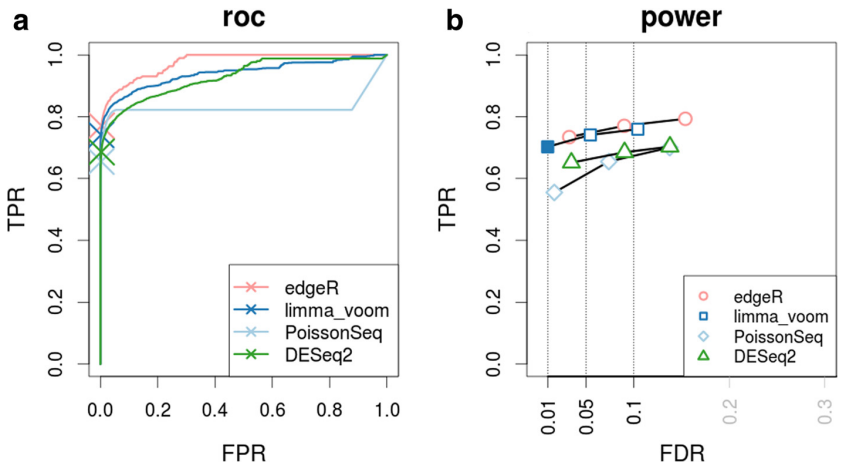

Fig. 3 Performance for zero-in-one-condition simulation. **a** ROC curves and **b** true positive rate (TPR) versus 'achieved' FDR curves of DE methods for the simulation dataset with zero-counts introduced as the true DE genes (overall performance of three simulations). The achieved FDR is the actual rate of false discoveries at the corresponding cutoff and this rate should ideally be controlled at the desired level. For the ROC curves, the *cross* on each curve represents the method's TPR at the (estimated) 5 % FDR cutoff. For the TPR versus achieved FDR curves, *points* are plotted at the following cutoffs: 0.01, 0.05 and 0.1. *Filled-in points* represent that the method has correctly controlled the error rate at the cutoff. *FDR* false discovery rate, *FPR* false positive rate, *roc* receiver operating characteristic, *TPR* true positive rate

Discussion

As developers and users of bioinformatics strategies, we are particularly interested in the metrics and methods that differentiate performance between the available tools. In this paper, we claim that count-based methods perform well when genes are only expressed in one condition, in contrast to an earlier report. We showed that a code error and the chosen scale of S/N resulted in the earlier conclusion that count-based methods suffer performance in this situation. By calculating the S/N on a different scale and using the same set of labels across methods, a reversal of method performance was observed. This highlights a sensitivity to decisions made in constructing the benchmark.

Using a customized simulation that introduces zero-counts in one experimental condition, we demonstrated that the performance of the count-based method is actually on a par with or better than other methods. We also debunked the postulation that poor performance is related to dispersion estimation in count models.

In the process of seeking the origins of this statistical performance difference, we discovered another potentially interesting phenomenon that may affect the interpretation of results. Looking at Table 1 and Additional file 1, it is evident that the scale of *P* values is drastically different between edgeR and voom. Although this observation appears rather unrelated to the ability to separate true from false DE genes, it is an indication that the scale of observations modeled affects the magnitude of statistical evidence derived. Not surprisingly, method performance is ultimately dependent on the scales, parameters and datasets used for the evaluation.

Software

R code and data that can be used to reproduce the figures in the main manuscript and in the supplement are available online [13].

Additional files

Additional file 1: Table of statistics for zero-count genes. Table of zero-counts, differential statistics and S/N for the ENCODE dataset. (CSV 112 kb)

Additional file 2: Supplementary figures. This file contains the mentioned supplementary figures. (PDF 712 kb)

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

Both MDR and XZ designed and conducted analyses and wrote the manuscript. Both authors read and approved the final manuscript.

Acknowledgments

XZ is supported by an Swiss National Science Foundation project grant (143883). MDR acknowledges funding from the European Commission through the 7th Framework Collaborative Project RADIANT (grant agreement number 305626). We acknowledge data processing help from Malgorzata Nowicka at the outset of the project and to her and Charity Law for helpful feedback on an earlier version of the manuscript.

Received: 11 December 2014 Accepted: 15 September 2015
Published online: 08 October 2015

References

1. Robinson MD, McCarthy DJ, Smyth GK. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics. 2010;26(1):139–40.
2. Anders S, Huber W. Differential expression analysis for sequence count data. Genome Biol. 2010;11(10):106. doi:10.1186/gb-2010-11-10-r106.
3. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology. 2014;15(12):550.
4. Law CW, Chen Y, Shi W, Smyth GK. Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. Genome Biol. 2014;15(2):29. doi:10.1186/gb-2014-15-2-r29.
5. Li J, Witten DM, Johnstone IM, Tibshirani R. Normalization, testing, and false discovery rate estimation for RNA-sequencing data. Biostatistics. 2012;13(3):523–38.
6. Smyth GK. Limma: linear models for microarray data. Chap. 23. In: Bioinformatics and computational biology solutions using R and Bioconductor. New York: Springer; 2005. p. 397–420. doi:10.1007/0-387-29362-0_23.
7. Rapaport F, Khanin R, Liang Y, Pirun M, Krek A, Zumbo P, et al. Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data. Genome Biol. 2013;14(9):95.
8. Ziller MJ, Gu H, Müller F, Donaghey J, Tsai LT-Y, Kohlbacher O, et al. Charting a dynamic DNA methylation landscape of the human genome. Nature. 2013;500(7463):477–81. doi:10.1038/nature12433.
9. Soneson C, Delorenzi M. A comparison of methods for differential expression analysis of RNA-seq data. BMC Bioinformatics. 2013;14(1):91. doi:10.1186/1471-2105-14-91.
10. Soneson C. compcodeR – an R package for benchmarking differential expression methods for RNA-seq data. Bioinformatics. 2014;30(17):2517–18.
11. Zhou X, Lindsay H, Robinson MD. Robustly detecting differential expression in RNA sequencing data using observation weights. Nucleic Acids Res. 2014. doi:10.1093/nar/gku310.
12. Montgomery SB, Sammeth M, Gutierrez-Arcelus M, Lach RP, Ingle C, Nisbett J, et al. Transcriptome genetics using second generation sequencing in a Caucasian population. Nature. 2010;464(7289):773–7.
13. Additional material. http://imlspenticton.uzh.ch/robinson_lab/zero_count/.
14. Huber W, von Heydebreck A, Sültmann H, Poustka A, Vingron M. Variance stabilization applied to microarray data calibration and to the quantification of differential expression. Bioinformatics. 2002;18 Suppl 1:96–104. doi:10.1093/bioinformatics/18.suppl_1.S96.

CHAPTER IV

miRNA-Seq normalization comparisons need improvement

Xiaobei Zhou, Alicia Oshlack and Mark D. Robinson

Paper published in *RNA (2013) 19 (6), 733-734*

miRNA-Seq normalization comparisons need improvement

XIAOBEI ZHOU,[1,2] ALICIA OSHLACK,[3] and MARK D. ROBINSON[1,2,4]

[1]Institute of Molecular Life Sciences, University of Zurich, CH-8057 Zurich, Switzerland
[2]SIB Swiss Institute of Bioinformatics, University of Zurich, CH-8057 Zurich, Switzerland
[3]Murdoch Children's Research Institute, Royal Children's Hospital, Parkville, Victoria 3052, Australia

BACKGROUND

Currently there is no method of best practice for the normalization of microRNA sequencing data (miRNA-Seq). Therefore, we read with interest a recent article in *RNA* by Garmire and Subramaniam that set out to compare various normalization strategies specifically for this application (Garmire and Subramaniam 2012). They compared methods currently in use for normalization of messenger RNA sequencing (mRNA-Seq) data, such as total-depth normalization ("raw") and Trimmed Mean of M-values ("TMM"). Additionally, they compared many methods not used previously with sequencing data, such as global scaling, and borrowed from strategies applied to microarray studies, such as quantile normalization (QN). The article attracted our attention for many reasons, but notably for the claimed poor performance and "abnormal results" of our TMM method (Robinson and Oshlack 2010). After investigating, we discovered that TMM's claimed poor performance was the result of an error that shifted log-ratios in the wrong direction. Furthermore, we felt that various practical issues were not satisfyingly discussed; we comment briefly on these here and provide reproducible reanalyses to support our claims (see Supplemental Material).

REPRODUCIBILITY

The authors were confused about how to introduce the TMM normalization factors (private e-mail to us November 6, 2010; code sent privately to us on August 3, 2012). While we did not answer this question directly in the original exchange, we pointed them to our online example code where the TMM normalization factors are introduced to the statistical test. Importantly, as mentioned in the TMM article (Robinson and Oshlack 2010), the normalization factors modify the *library size*, not the *count data*. Therefore, Garmire and Subramaniam's abnormal TMM results can be attributed to introducing these factors in the wrong direction (see Supplemental Note S1 for the correction). We make our R code publicly available, so others can reproduce our analyses and test new situations; documentation for applying TMM in a standard setting is readily available in the edgeR software package (Robinson et al. 2010). However, it is the user's responsibility to ensure correct usage in a nonstandard setting (e.g., operations on log-ratios instead of differential expression statistics).

We have reproduced some of the metrics presented in the Garmire and Subramaniam paper and conclude that the corrected TMM normalization is an average performer and represents an improvement over total-depth normalization (Supplemental Note S2). However, the integration of TMM normalization factors within an established statistical framework provides a clear path from raw data to interpretable statistical summaries (e.g., *P*-values), whereas other methods (e.g., QN) may not, at least in small samples where parametric models are used. Therefore, we question the validity of some of Garmire and Subramaniam's comparisons and also the overall conclusions of the paper, as discussed below.

MSE AND K-S METRICS ARE NOT APPROPRIATE IN THIS SETTING

The purpose of normalization is to remove technical bias while maintaining true biological signal. Garmire and Subramaniam employed mean-squared error (MSE) and the Kolmorogov-Smirnov (K-S) test metrics, among others, to assess normalization performance. A small MSE or K-S statistic, applied here to single samples from *different* biological conditions, was taken by Garmire and Subramaniam to be evidence of good performance. Unfortunately, this comparison gives no consideration to the presence of truly differentially expressed miRNAs, which directly affect these scores. Low MSE favors normalization that removes all evidence of differential expression, which is an undesirable property when true biological differences exist (e.g., here, evidence from corresponding miRNA qPCR data). Notably, the cited reference that uses MSE as a performance metric does so from known (simulated) fold-changes (Xiong et al. 2008). A more appropriate performance metric would be MSE or scale-free coefficient of variation between biological *replicates* of the same condition, as recently reported for comparing mRNA-Seq normalization strategies (Dillies et al. 2012).

[4]**Corresponding author**
E-mail mark.robinson@imls.uzh.ch
Article published online ahead of print. Article and publication date are at http://www.rnajournal.org/cgi/doi/10.1261/rna.037895.112.

The K-S test measures the similarity of two cumulative distributions. We question the motivation for this, at two levels: (i) Samples with different "composition" exhibit different marginal distributions (e.g., comparisons of kidney and liver tissue; Supplemental Fig. S8 in Additional file 1 of Robinson and Oshlack 2010); and (ii) QN would always achieve a zero K-S statistic, were it not for the treatment of ties (Supplemental Note S3). Therefore, QN is always put in a favorable light by this comparison, regardless of any nonlinear effects introduced.

It is worth noting that Garmire and Subramaniam's performance comparisons disregard features that are unobserved in one of the two conditions (i.e., count of zero), since fold-changes cannot be computed. However, miRNAs present in one condition and absent in another may be biologically interesting and should not be ignored, which calls into question how to apply QN in practical situations and whether these performance comparisons are representative of the whole data set. Discarding data for the purposes of performance evaluation may be permissible, but removing such data in downstream analyses is clearly undesirable.

STATISTICAL METHODS FOR COUNT DATA NEED COUNTS

As mentioned, TMM preserves the count data by introducing normalization factors as *offsets* in the statistical model (Robinson and Oshlack 2010). In contrast, Garmire and Subramaniam proceeded to use count-based statistical tests (Fisher exact, Binomial, Poisson, and χ^2) to normalized non-count data. We have two reservations about this approach: (i) The tests employed do not have the capacity to address biological variability, which is essential to generalizable conclusions (Hansen et al. 2011); (ii) transforming count data into nonintegers can distort the mean-variance relationships implied by existing count models (Oshlack and Wakefield 2009). Regardless, clear recommendations of how to apply normalization in a practical setting are needed.

REFERENCE DATA SETS

In order to make decisive claims about method performance, "reference" data sets are critical. Such data sets include an independent truth (e.g., measurements from an independent platform) that can be used to evaluate the performance of an algorithm. Garmire and Subramaniam employed receiver operator characteristic (ROC) curves using miRNA qPCR as the independent truth to define truly differential (and non-differential) miRNAs. Our reanalyses of this data set suggest that ROC results are sensitive to decisions made in determining the "truth" (Supplemental Note S4). Altogether, we conclude that the ROC analysis performed by Garmire and Subramaniam is not conclusive, without a further sensitivity analysis of parameters affecting the selection of true positive and true negatives.

SUMMARY

As developers and users of informatics strategies, we are keenly interested in the relative merits of competing approaches. Crucially, there has been relatively little investigation into normalization strategies for miRNA-Seq data and the timely article from Garmire and Subramaniam promised to shed light on this issue. Unfortunately, errors in the implementation, poor choice of performance metrics (or poor choice of data set), few details about practical implementation (e.g., elimination of features containing zero count), and sensitivity to choices made regarding the reference truth data set have left many open questions about the best analysis methods for miRNA-Seq data. In this paper, we have discussed some of the subtle yet critical parameters that need to be carefully investigated.

SUPPLEMENTAL MATERIAL

Supplemental material is available for this article.

ACKNOWLEDGMENTS

X.Z. is supported by SNSF project grant (143883). A.O. is supported by an NHMRC career development fellowship (ID: 1051481). M.D.R. acknowledges funding from the European Commission through the 7th Framework Collaborative Project RADIANT (Grant Agreement Number: 305626).

REFERENCES

Dillies M-A, Rau A, Aubert J, Hennequet-Antier C, Jeanmougin M, Servant N, Keime C, Marot G, Castel D, Estelle J, et al. 2012. A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Brief Bioinform* doi: 10.1093/bib/bbs046.

Garmire LX, Subramaniam S. 2012. Evaluation of normalization methods in mammalian microRNA-Seq data. *RNA* **18:** 1279–1288.

Hansen KD, Wu Z, Irizarry RA, Leek JT. 2011. Sequencing technology does not eliminate biological variability. *Nat Biotechnol* **29:** 572–573.

Oshlack A, Wakefield MJ. 2009. Transcript length bias in RNA-seq data confounds systems biology. *Biol Direct* **4:** 14.

Robinson MD, Oshlack A. 2010. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biol* **11:** R25. doi: 10.1186/gb-2010-11-3-r25.

Robinson MD, McCarthy DJ, Smyth GK. 2010. edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26:** 139–140.

Xiong H, Zhang D, Martyniuk CJ, Trudeau VL, Xia X. 2008. Using Generalized Procrustes Analysis (GPA) for normalization of cDNA microarray data. *BMC Bioinformatics* **9:** 25.

miRNA-Seq normalization comparisons need improvement

Xiaobei Zhou, Alicia Oshlack and Mark D. Robinson

RNA 2013 19: 733-734 originally published online April 24, 2013
Access the most recent version at doi:10.1261/rna.037895.112

CHAPTER V

Discussion and perspectives

1 Discussion

1.1 Robust M-Estimator or weighted likelihood estimator for robust method?

As mentioned previously (see Section 2.6.1), the robust M-estimator would be difficult to implement within the current weighted APL framework of ***edgeR***. Skipping the difficulty of implementation, robust M-estimator should be valuable in count-based methods based on the GLM NB model. Robust M-estimator motivated by robust statistics conventionally fit data contaminated with outliers. William et al. introduced an robust NB model based on M-estimator [1]. They discussed statistical inference (i.e., asymptotic normality of estimated coefficients and dispersion) of the robust NB model. In fact, the asymptotic and robust properties of weighted likelihood estimator has been proved by researchers [19, 20]. In addition, Markatou et al. compared these two estimators and made conclusions that the performances of weighted likelihood estimator are exactly equivalent with robust M-estimator in most of their simulations [18].

1.2 Could poor FDR control be linked to the inappropriate observation weights of our robust method?

In Chapter I, our robust method has been shown a somewhat liberal performance that suffers a poor FDR control, particularly for the simulations in absence of outliers. The link between the liberal results and the inappropriate observation weights used in the robust method in some case should exist. This might be explained by a naive simulated test: when the truth (outliers) are known and weights of truth are assigned as 0, our robust method could perfectly dampen the effect of the outliers (this has been tested in our simulation framework). One possible explanation is that our robust method lacks a convergence mechanism. Sometimes weights are not appropriately assigned due to not achieving convergence during the process of iteratively reweighted least squares. In fact, in real case it is difficult to know exactly what an outlier is and if it is how much it should be down-weighed. A little liberal result of the robust method is acceptable compromise to achieve a good tradeoff statistical power and protection against outliers. Beyond the robust approach, an alternative solution in case of data contaminated with outliers is using Quasi-likelihood inference. However, Quasi-method [17] seems very conservative in "normal" data without outliers based on the results of our count-based simulations.

2 Perspectives

2.1 My contribution to DTE analysis

2.1.1 General framework settings

Before representing the models that we developed for DTE analysis, some general notation is firstly introduced.

For feature (e.g., transcript) i, the original sample (OS) with size of n is denoted as:

$$\begin{bmatrix} y_{i1} & y_{i2} & y_{i3} & \dots & y_{in} \end{bmatrix}, \tag{1}$$

where y_{ij} is the count data for feature i in OS j. Its corresponding bootstrapping samples (BSs) for feature i (assuming that each experimental unit has the same number of bootstraps m) are:

$$\begin{bmatrix} \dot{y}_{i11} & \dot{y}_{i21} & \cdots & \dot{y}_{ij1} & \cdots & \dot{y}_{in1} \\ \dot{y}_{i12} & \dot{y}_{i22} & \cdots & \dot{y}_{ij2} & \cdots & \dot{y}_{in2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \dot{y}_{i1k} & \dot{y}_{i2k} & \dot{y}_{i3k} & \dot{y}_{ijk} & \cdots & \dot{y}_{ink} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \dot{y}_{i1m} & \dot{y}_{i2m} & \cdots & \dot{y}_{ijm} & \cdots & \dot{y}_{inm} \end{bmatrix}, \tag{2}$$

where $\dot{y}_{ijk}$ is the kth bootstrap-estimated count for feature i in BS j. There are two definitions that have to be clarified: i) The jth (i.e., j sample) BS (colored green) for feature i is:

$$\begin{bmatrix} \dot{y}_{ij1} & \dot{y}_{ij2} & \cdots & \dot{y}_{ijk} & \cdots & \dot{y}_{ijm} \end{bmatrix}^T, \tag{3}$$

denoted as $\dot{Y}_{ij\circ}$. ii) The kth subsample of BSs (colored blue) for feature i is:

$$\begin{bmatrix} \dot{y}_{i1k} & \dot{y}_{i2k} & \dot{y}_{i3k} & \cdots & \dot{y}_{ink} \end{bmatrix}, \tag{4}$$

denoted as $\dot{Y}_{i\bullet k}$. **Note** that there is no association between the values $\dot{y}_{.1.}$, $\dot{y}_{.2.}$ and $\dot{y}_{.m.}$ in the same subsample.

2.1.2 Ensembles

Ensembles can improve the predictive performance and robustness of certain estimators or algorithms, such as linear models with variable selections and the classification and regression tree (CART) algorithm. The motivation of using ensembles for DTE analysis is to help count-based methods handle the uncertainty of estimating transcript abundance by combining multiple estimators (e.g., using estimates from bootstraps) to reduce variability. As we know, the standard NB model cannot directly handle this uncertainty. The NB mixed model can be used to handle the uncertainty but it lacks an appropriate shrinkage or moderation procedure to improve variance (or dispersion) estimation accuracy. The strategy of ensembling individual predictions of the count-based method for each bootstrapping subsample is therefore an alternative solution for DTE analysis. The basic idea of the ensembled NB estimator for DTE analysis is to ensemble single statistics (e.g., likelihood ratio) of BS subsamples fitted by the NB model together to obtain an overall result. The detailed information of ensemble methods for DTE analysis is introduced in the next sections.

Bagging of (*edgeR*) likelihood ratios

Bootstrap aggregating (bagging) was initially used to improve the predictive performance of tree models. The basic idea of bagging is that of averaging the prediction of individual models with an equal weight to form a final prediction. For DTE analysis, bagging of likelihood ratios (LRs) between the null and alternative hypotheses is presented as following:

$$\widehat{LR}_{bag} = m^{-1} \sum_{k=1}^{m} LR_{i\bullet k}, \tag{5}$$

where $LR_{i\bullet k}$ is the likelihood ratio of the kth subsample of BSs for feature i and m is the number of bootstrap samples. Here the LRs can come from either ***edgeR*** or other DGE methods, which can provide LRs of data between the null and alternative hypotheses. In fact, this bagging approach is not limited to LR tests but also applies to any method that can provide an appropriate score or statistic, such as a Wald test or T test statistic.

One major disadvantages of bagging is that it is a biased estimator because bagging would increase the bias by reducing the variance during a bias-variance tradeoff [5, 6]. Whether a bias-variance tradeoff exists in case of bagging of LRs for DTE analysis is not clear. Additionally, if a tradeoff existed, the biased estimator of bagging LRs would be potentially harmful in case of low or no uncertainty of transcript abundance estimation. Another disadvantage is that it is computationally intensive for a large number of bootstrapping samples.

Combination of p-values

For DTE analysis, another ensemble method is to directly combine multiple p-values of an subsample of BSs for the same feature to compute an overall p-value. An existing model framework, which works for the statistical inference for a group of genes (e.g., gene ontology terms) introduced by Delongchamp et al. [8], can be directly used for DTE analysis. In fact, this model borrows ideas from meta-analysis methods for combining p-values.

In the case of a DTE analysis, $p_{i\bullet k}$, the p-value of the kth SB subsample for feature i, is assumed to follow a uniform distribution under the null hypothesis. Then we get a transformation of $p_{i\bullet k}$ following the standard normal distribution:

$$z_{i\bullet k} = \Phi^{-1}(1 - p_{i\bullet k}). \tag{6}$$

If the set of p-values $\{p_{i\bullet k} : k = 1, \ldots, m\}$ are independent, the form $m^{-1/2} \sum_{k=1}^{m} z_{i\bullet k}$ converges to a standard normal distribution according to the central limit theorem (CLT). Thus the overall significance, p_i equals:

$$\hat{p}_i = 1 - \Phi \left\{ \sum_{k=1}^{m} z_{i\bullet k} / \sqrt{m} \right\}. \tag{7}$$

In fact, subsample of BSs are highly correlated. Ignoring correlations would overstate the overall statistical significance and therefore an adjustment for correlation is necessary. The method combining a set of p-values into an overall significance level with an adjustment for correlation is presented as follows:

$$\hat{p}_i = 1 - \Phi \left\{ \frac{1^T Z_i}{\sqrt{1^T R_i 1}} \right\}, \tag{8}$$

where Z_i is $(z_{i\bullet 1}, ..., z_{i\bullet m})^T$ and 1 is $(1, 1, ..., 1)^T$. The covariance of Z_i is R_i. The difficulty here is to accurately estimate the covariance R_i when the treatment effect of each BS subsample

appears as a confounding factor. A possible solution is to simplify R_i as a constant term r considered as a random block effect and estimate r by the existing model framework in ***limma*** ("duplicateCorrelation").

2.1.3 Parametric bootstrap

In the situation where current transcript quantification tools can provide an unlimited number of BS subsamples, there are two major advantages of parametric bootstrap applied for DTE analysis: i) parametric bootstrap (unlike permutation) allows a flexible design matrix, and ii) parametric bootstrap does not rely on the approximation of the null distribution (e.g., χ^2). It will improve performance in the condition where the uncertainty of transcript abundance estimation affects the null distribution (this needs further research).

For DTE analysis, parametric bootstrap can be summarized as:

i) Generate a subsample $\tilde{Y}_{i\bullet k}$ by simulating from $\hat{f}_0(\dot{Y}_{i\bullet k})$, where $\hat{f}_0$ denotes the fitted distribution under the null hypothesis and $\dot{Y}_{i\bullet k}$ is the kth BS subsamples for feature i.
ii) Calculate $\tilde{LR}_{i\bullet k}$ for $\tilde{Y}_{i\bullet k}$.
iii) Repeat i) and ii) m times.
v) Calculate LR_i for Y_i from the OS.
vi) Calculate p-value$= \frac{n_{extreme}+1}{m+1}$, where $n_{extreme} = \sum_{K=1}^{m} I(\widetilde{LR}_{i\bullet k} > LR_i)$.

The disadvantage of parametric bootstrap for DTE analysis is the computational cost.
Another application of parametric bootstrap for DTE analysis is to adjust the LR statistic by a Bartlett correction [7, 11, 13]. The adjusted LR_i with a Bartlett correction factor estimated from parametric bootstrap is:

$$\widehat{LR}_i = \frac{LR_i}{E_{\widetilde{LR}_i\bullet k}/d}, \tag{9}$$

where $E_{\widetilde{LR}_i\bullet k} = \sum_{k=1}^{m} \widetilde{LR}_{i\bullet k}$ and d is the degrees of freedom of the statistic test.

2.1.4 Meta random-effects regression model

Meta random-effects regression model

In contrast to ensemble methods that combine multiple estimators of a BS subsample for DTE analysis to account for estimation variability, we also take interest in building a model that can directly propagate the variances of BSs into the statistical inference. For example, a random-effects model for meta-analysis introduced by Berkey can be applied for DTE analysis [4]. Compared to (classical) linear models, this model has an additional covariance component structure allowing for different sources of variability. For DTE analysis, we assume that the individual sample error v_{ij} of counts y_{ij} for feature i $(i = 1, \ldots, z)$ in OS j $(j = 1, \ldots, n)$, which represents the uncertainty of estimating of transcript abundance, can be estimated from the jth BS $\dot{Y}_{ij\circ}$. The meta random-effects model with individual sample errors in the case of a DTE analysis is presented as follows:

$$\mu_{ij} = X\beta_i + v_{ij} + \epsilon_i, \tag{10}$$

where X is the design matrix. The response variable μ_{ij} could be log-counts per million (log-cpm), as used by ***limma-voom*** [15], which is defined as:

$$\mu_{ij} = log_2(\frac{y_{ij} + 0.5}{L_j + 1}) \tag{11}$$

where L_j is the (normalized) library size for OS j. This transformation is necessary, since count data should not be directly used in linear models due to its strong mean-variance relationship. v_{ij} is the sample-specific estimation uncertainty error that is assumed to be $\mathcal{N}(0, \sigma_{ij}^2)$ distributed, and ϵ_i is a random error to represent the (pure) feature biological variation assumed to follow:

$$\mathcal{N}(0, \tau_i^2). \tag{12}$$

The regression coefficients, β_i, can be estimated by a weighted-least-squares (WLS) estimation:

$$\hat{\beta}_i = ({X_i}^T W_i X_i)^{-1} X^T W_i U_i, \tag{13}$$

where $U_i = diag(u_{i1}, ..., u_{in})$. and $W_i = diag(w_{i1}, ..., w_{in})$. Each diagonal component of weights, w_{ij}, equals

$$w_{ij} = \frac{1}{\hat{\tau}_i^2 + \hat{\sigma}_{ij}^2}\ , \tag{14}$$

where $\hat{\tau}_i$ is the estimation of τ_i and $\hat{\sigma}_i$ is the estimation of σ_{ij}.
To estimate τ, there are numerous methods suggested in the meta analysis literature [12, 23, 25]. Here for DTE analysis, the Hedges estimator [12] is employed, since it is easy to implement and less computationally intensive. In fact, the idea of Hedges estimator is equivalent to feasible generalized least squares (FGLS) estimator in political science, introduced by Lewis et al. [16]. The estimator for τ_i can be constructed as:

$${\hat{\tau}_i}^2 = \frac{\sum_j^n r_{ij}^2 - \sum_j^n \sigma_{ij}^2 + tr((X^T X)^{-1} X^T V_i X)}{n - p}, \tag{15}$$

where p is the dimension of the full-model parameter space, r_{ij} is the residual of ordinary least square (OLS) regression without weights and V_i is a $n \times n$ diagonal matrix with σ_{ij}^2 as the jth diagonal element.

The σ_{ij}^2 are assumed to be known for this model. In the case of a DTE analysis, σ_{ij}^2 can be estimated from BS datasets with sufficient accuracy, depending on the number of BSs. We can estimate σ_{ij}^2 as

$$\hat{\sigma}_{ij}^2 = var(\dot{M}_{ij\circ}) \tag{16}$$

where $\dot{M}_{ij\circ}$ is the log-cpm of jth BS for feature i. Practically, we found that certain $\hat{\sigma}_{ij}$ obtained from BSs produced by ***sleuth*** are extremely large ($\geqslant 10$). This would affect the accuracy of estimates of biological variance τ_i in the meta random-effects regression model. We suggest

to trim the top 5 percentage of $\hat{\sigma}_{ij}$ as a constant threshold value to dampen their effects.

To increase power to detect differential features, it is necessary to have an appropriate moderation/shrinkage procedure to moderate/shrink the variance/dispersion parameters towards a prior distribution or trend. Here, the methodology to estimate the mean-variance relationship is the same as in ***limma-voom***: a smooth LOWESS curve fitting the square-root standard deviations $\hat{\tau}_i^{1/2}$ as a function of mean log-counts $\tilde{\lambda}_i$. $\tilde{\lambda}_i$ can be calculated as:

$$\tilde{\lambda_i} = \sum_{j=1}^{n} \frac{\hat{\lambda}_{ij}}{n}, \tag{17}$$

where $\hat{\lambda}_{ij} = \hat{\mu}_{ij} + log_2(L_j) - log_2(10^6)$ and $\hat{\mu}_{ij} = x_{ij}\beta_i$.

The LOWESS curve is used to define a function $l()$ to shrink variance towards a trended-by-mean estimate. The shrinkage estimate of biological standard deviation is:

$$\tilde{\tau}_i = l(\hat{\lambda}_{ij}). \tag{18}$$

However, based on prior experience from the DE method study [2], we know that the shrinkage estimates cannot represent all the variability of features. For certain features, their variability are much larger than the predicted value of fitted LOWESS curve. We use a similar strategy as ***DESeq*** [2] to process the shrinkage estimate: if the shrinkage estimate is less than the LOWESS curve, we shift it to the curve; if the shrinkage estimate is larger than the curve, we keep it as is. The modified shrinkage estimate of biological standard deviation can be obtained as:

$$\hat{\tilde{\tau}}_i = pmax(\tilde{\tau}_i, \hat{\tau}_i). \tag{19}$$

Then the shrinkage weights $\hat{w}_{ij}$, which equals $\frac{1}{\hat{\tilde{\tau}}_i^2 + \hat{\sigma}_{ij}^2}$, is propagated into the WLS regression to estimate final regression coefficients:

$$\hat{\tilde{\beta}}_i = ({X_i}^T \hat{W}_i X_i)^{-1} X^T W_i U_i, \tag{20}$$

where $\hat{W}_i = diag(\hat{w}_{i1}, ..., \hat{w}_{in})$.

Additionally, moderated t-statistic [27] based on an empirical Bayesian approach can be used in this meta random-effects regression model to increase the detection power. The key point of the moderated t-statistic is that a posterior estimate of the variance, instead of individual sample variance, is used. However, there is one difficulty to directly apply this statistic in the meta random-effects regression model. It has multiple variance components τ_i and σ_{ij}, while the moderated t-statistic that can squeeze the sample variances towards a common value, or to a global trend based on an empirical Bayesian algorithm is designed to work for models with only one variance component, such as OLS model.

We use an approach from Knapp et al. [14] to calculate a single pseudo-variance by summing the WLS residuals:

$${\gamma_i}^2 = \frac{\sum_{j=1}^{n} w_{ij}(\mu_{ij} - \hat{\mu}_{ij})^2}{d_i}, \tag{21}$$

where d_i is the degrees of freedom (d.f.) of the meta regression model. Then the posterior pseudo-variance is estimated by existing methodology in ***limma***:

$$\hat{\gamma_i}^2 = \frac{d_0{\gamma_0}^2 + d_i{\gamma_i}^2}{f_0 + d_i} \ , \tag{22}$$

where γ_0 is the prior estimated from the marginal distribution and d_0 is its estimated d.f.. The final moderated t-statistic of meta random-effects regression model for DTE analysis is:

$$t_{ij} = \frac{\hat{\hat{\beta}}_{ij}}{c_{ij}\hat{\gamma}_i} \ , \tag{23}$$

where $\hat{\hat{\beta}}_{ij}$ is the estimated coefficient of WLS regression using the shrinkage weights (Equation 20) and c_{ij} is the unscaled standard deviation. In this case, c_{ij} equals the jth diagonal element of $({X_i}^T\hat{W}_iX_i)^{-1}$, where $\hat{W}_i = diag(\hat{w}_{i1}, ..., \hat{w}_{in})$. Here, the single elements $\hat{w}_{ij}$ are the shrinkage weights but not the original weights; and this approach is similar to ***limma-voom***.

Comparison between meta random-effects regression model and *sleuth*

sleuth is similar in many respects to this meta random-effects regression model. Several distinctions between them are highlighted below:

Firstly, meta random-effects regression model can precisely handle multiple sample errors of estimation, while the model in ***sleuth*** can only work for one single sample error. The current model in ***sleuth*** can be considered as a simplified version of the meta regression model (Equation 10) with a single sample error, v_i, to represent an overall level of variability of transcript estimation across all the samples:

$$\mu_{ij} = X\beta_i + v_i + \epsilon_i, \tag{24}$$

where $v_i \sim \mathcal{N}(0, \sigma_i^2)$ and $\epsilon_i \sim \mathcal{N}(0, \tau_i^2)$. Again, ϵ_i represents the biological variability, σ_i^2 is the technical variance of transcript estimation and τ_i^2 is the biological variance and$X$ is the design matrix.

Secondly, our meta random-effects regression model uses a modified version of the moderated t-statistic, while ***sleuth*** uses the likelihood ratio test (LRT) under the full and reduced model. During the LRT, an estimate of total variance D_i is used to generate the likelihood in ***sleuth*** [21]. D_i can be expressed in the form of:

$$D_i = max(\hat{\tilde{\tau}}_i^2, \hat{\tau}_i^2) + \hat{\sigma}_i^2 \tag{25}$$

where $\tilde{\tau}_i^2$ and $\hat{\tau}_i^2$ are the shrinkage and raw estimate of biological variance and $\hat{\sigma}_i^2$ is the estimate of technical variance of transcript estimation.

Finally, meta random-effects regression model is based on the WLS regression using the observation weights to handle inconsistency of sample errors (i.e., heteroscedasticity), while ***sleuth*** uses a variance component structure that separates the two sources of variance: σ_i^2 and τ_i^2.

2.1.5 Modified NB model with additional dispersion parameter to handle the uncertainty in count data

Compared to a standard NB model, an ideal modified version of NB model that can precisely account for additional variability of transcript estimation of each biological sample could be represented by the following formula:

$$Y_{ij} \sim NB(\mu_{ij}, \phi_i + \psi_{ij}), \tag{26}$$

where y_{ij} is the count data for feature i in OS j, μ_{ij} is the mean for feature i in OS j, ϕ_i is the dispersion for feature i related to (pure) biological variation across samples, denoted as biological dispersion; and ψ_{ij} is the sample-specific dispersion attributable to transcript abundance estimation, denoted as inferential dispersion. The variance of y_{ij} can be represented in the form of:

$$var(y_{ij}) = \mu_{ij} + (\phi_i + \psi_{ij})\mu_{ij}{}^2 \tag{27}$$

However, this model with multiple inferential dispersions cannot be easily implemented in the current model framework of ***edgeR*** due to its complex structure while moderating the dispersion estimates. Similar to the strategy of ***sleuth***, we simplify this model using a single dispersion ψ_i that relates to the average variability of transcript abundance estimation across samples instead of multiple inferential dispersions. The simplified model is presented as:

$$Y_{ij} \sim NB(\mu_{ij}, \phi_i + \psi_i). \tag{28}$$

In the settings of this model, inferential dispersion ψ_i is feature-specific but not sample specific as before. It can be estimated from BSs provided by transcript quantification tools (e.g., ***kallisto***). For feature i in the jth OS, we can get an estimate of dispersion $\hat{\psi}_{ij}$, which represents technical variation for this OS assuming that all the bootstrapping counts for this OS are considered technical replicates. The estimate of ψ_i equals $\frac{\sum_{j=1}^{n} \hat{\psi}_{ij}}{n}$.

Biological dispersion ϕ_i can be estimated by the usual methodology of the standard NB model framework given an estimated ψ_i assumed known up to a constant. The current existing successful model framework from ***edgeR*** is used to estimate ϕ_i. APL (see Section 1.3.2) can be employed as a dispersion estimator to reduce the bias introduced by maximum likelihood estimation (MLE) in presence of a nuisance parameter (regression parameter β); and moderation methodology (see Section 1.3.3) can be used to improve power (i.e., dispersion estimation precision) based on moderating dispersion towards a trended-by-mean estimate via weighted likelihood.

However, a situation, where inferential dispersion ψ affects the result of biological dispersion (ϕ) estimate, requires additional attention. Generally speaking, the estimate of ϕ_i without any moderation (0 prior d.f.) would be smaller than the result from the standard NB model in ***edgeR*** ignoring information of $\hat{\psi}_i$. Particularly, when $\hat{\psi}_i$ is extremely large, the estimated ϕ_i would become 0. One example of comparison between the estimate of ϕ of the NB model ignoring/considering ψ is shown in Figure 1. The effect of estimates under high expression value of ψ ($\psi > 5$) are colored blue. The sets of high expressed ψ would obviously affect the variance-mean relationship fitting count data and then affect later moderation procedure. Our

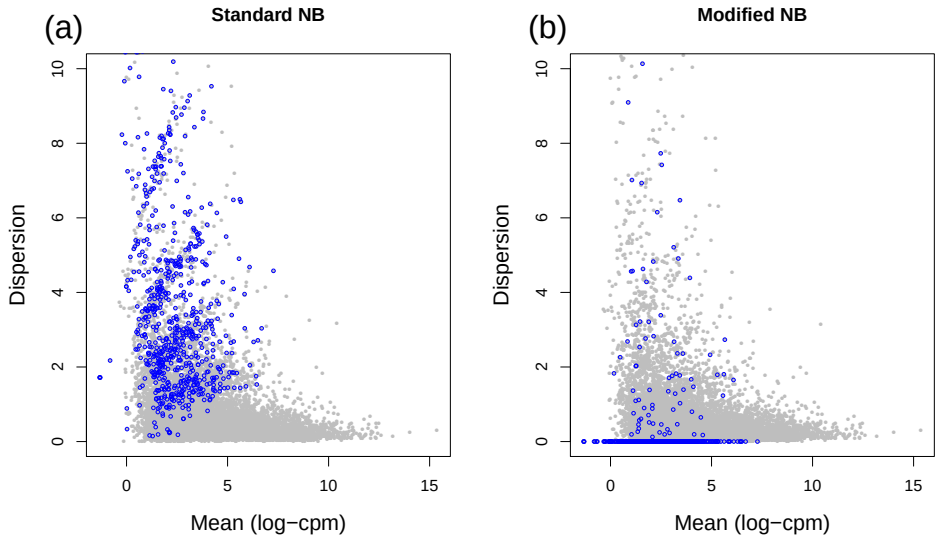

Figure 1.: The effect of inferential dispersion ψ attributable to transcript abundance estimation on the result of estimate of biological dispersion ϕ. The scatterplot of biological dispersion against average of transcript abundance (log2-cpm) for the standard/modified NB model in ***edgeR*** (ignoring/considering inferential dispersion) are shown on (a) and (b). The scatter points under high expression value of ψ ($\psi > 5$) are colored blue. The dataset coms from a simulation containing a two group (4 versus 4) comparison.

solution is to dampen the effect of ψ via feature-level weights. We make a modification to the existing moderation methodology as follows:

$$\begin{aligned} \phi_i &= \arg\max \left\{ APL_i(\phi_i) + \alpha \cdot \dot{APL}_{trend}(\phi_i) \right\} \\ &= \arg\max \left\{ APL_i(\phi_i) + \alpha \cdot \frac{\sum_{a \in C_i} w_a APL_a(\phi_i)}{\sum_{a \in C_i} w_a} \right\}, \end{aligned} \tag{29}$$

where α is the prior d.f. afforded to the shared likelihood and C_a is local shared set that is close to feature i in average log counts per million. The weights, w_a, can be calculated by the Huber function:

$$w_a = w(r_a) = \begin{cases} \frac{k}{abs(r_a)} & \text{for abs}(r_a) > k \\ 1 & \text{for abs}(r_a) \leq k, \end{cases} \tag{30}$$

where k is set to $1.345\hat{\sigma}$, $\hat{\sigma} = MAR/0.6745$ and MAR is the median absolute of r_a. $r_a = \dot{\phi}_a - \tilde{\phi}_a$ is the difference between the estimate of unmoderated dispersion with and without given information of estimated ψ_a. The approach to estimate σ is taken from the robust regression literature [10].

2.1.6 Generalized linear mixed models

For DTE analysis, a generalized linear mixed model (GLMM), as an extension of GLM allowing linear predictors to contain random effects addition to fixed effects, also seems natural to handle the uncertainty of transcript abundance estimation. For modeling the estimation error in the case of DTE analysis, it requires a GLMM could process the technical variation estimated from BSs (offered by the quantification tool) as random-effects and propagate them

into the regression model whose fixed-effects are fitted from OSs. Unfortunately, current existing GLMM frameworks could not satisfy this requirement [3, 22, 26]. Moreover, standard GLMM frameworks lack a shrinkage or moderation procedure that is necessary for fitting read counts to improve variance estimation accuracy. Additionally, the marginal density of the NB model (as a *de facto* model for genome-scale count data) under a GLMM framework cannot be expressed as a closed form [26]. This requires an approximation approach to the log likelihood resulting in more difficulties in implementing the moderation procedure. According to these, we develop a modified version of NB mixed model under the GLMM framework using Bayesian inference (with help from INLA [24]). In our setting, Y_{ij}, which are the read counts in sample j for feature i, are assumed to follow a NB distribution with mean μ_{ij} and dispersion ϕ_i, denoted by $Y_{ij} \sim NB(\mu_{ij}, \phi_i)$. The GLMM formula for fitting Y_{ij} with a canonical logarithm link is:

$$log(\mu_{ij}) = X\beta_i + Z\gamma_i + \log N_j, \tag{31}$$

where μ_{ij} is the mean response variable $(j = 1, \ldots, n)$ for feature i, β_i is the $p \times 1$ vector of fixed-effects, X is the $n \times p$ design matrix for the fixed effects, γ_i is the $q \times 1$ vector of random-effects, Z is the $n \times q$ design matrix for the random-effects related and N_j is the (effective) library size for sample j.

The random-effects, γ_i, are assumed to follow a Normal distribution with mean 0 and variance matrix G_i:

$$\gamma_i \sim \mathcal{N}(0, G_i^2). \tag{32}$$

In general settings of GLMMs, referring to the random-effects, the column of Z and structure of G can be specified according to the factors that the researchers are interested in (e.g., experimental conditions, batches, samples). In the case of a DTE analysis, we want to use observation-level random-effects to model the sample-specific variability of transcript estimation. We only focus on the random-effects related to the variability of transcript abundance estimation; and other factors are not considered into the model although they may be related to the random effects. Hence we can define Z as a $n \times n$ diagonal matrix with the same diagonal element and γ_i as a $n \times 1$ vector. G_i becomes a diagonal matrix, $diag(\sigma_{i1}^2, \ldots, \sigma_{in}^2)$, with σ_{ij}^2 as the jth diagonal element representing the variability of transcript abundance estimation in OS j for feature i. Practically, we assume that $\sigma_{i1} = \cdots = \sigma_{in} = \sigma_i$ for easier implementation and reduction of the computational cost.

As mentioned above, since the marginal density of this model is not a closed form, we apply Integrated Nested Laplace Approximation, ***INLA***, to generate its marginal density. For the Bayesian inference, we have to redefine the regression parameters in form of prior distributions. Each fixed-effects regression coefficient, β_{ij}, is assumed to follow a flat prior distribution:

$$\beta_{ij} \sim \mathcal{N}(0, 100). \tag{33}$$

Each random-effects regression coefficient, γ_{ij}, is assumed to follow:

$$\gamma_{ij} \sim \mathcal{N}(0, \hat{\sigma}_i^2), \tag{34}$$

where $\hat{\sigma}_i$ is the estimated σ_i. How to estimate σ_i will be discussed later. The dispersion prior, ϕ_i, is assumed to follow a Gamma distribution with a sparse shape that makes the variance

close to zero and the mean equal the estimated ϕ_i

$$log(\phi_i) \sim \Gamma(100000 * \hat{\phi}_i, \frac{1}{100000}), \tag{35}$$

where $\hat{\phi}_i$ is the moderated estimate of dispersion obtained from ***edgeR***.
Here ***INLA*** plays a core role to generate the marginal likelihood of the NB mixed model. Given estimated ϕ_i and σ_i, ***INLA*** can calculate the posterior inference of full and null models using Laplace approximation. Then the marginal likelihoods under full and null models where the random-effects are integrated out can be computed. Notably, ***INLA*** can easily compute the marginal likelihood by Laplace approximation when its integral expression is not a closed form. The basic (R-code) pipeline fitting count data to the NB mixed model under the GLMM framework using ***INLA*** is presented in the following lines:

```
form = mu~x+f(x,model="iid",initial=log(prec),fixed=TRUE)+offset(o)
fit = inla(form,data=dat,family="nbinomial",
    control.family=list(hyper=list(prior="loggamma",
    param=c(phi *(1e+5),1e+5))), ...)
l <- exp(fit$mlik[1])
```

Note that $prec = \frac{1}{\hat{\sigma}_i}$, $phi = \hat{\phi}_i$ and $\boldsymbol{l}$ is the marginal likelihood of the model.

Rather than providing a significance value (e.g., p-value) of a feature (e.g., gene) by the frequency-based model, a false discovery rate can be directly calculated by the Bayesian model framework without any further adjustment. We use the local false discovery rate introduced by Efron et al. [9] to measure differential expression level of a feature:

$$lfdr_i = \frac{p_0 ML(y_i, \mathcal{M}_0)}{p_0 ML(y_i, \mathcal{M}_0) + (1 - p_0) ML(y_i, \mathcal{M}_1)}, \tag{36}$$

where $ML(y_i, \mathcal{M}_0)$ and $ML(y_i, \mathcal{M}_1)$ are marginal likelihoods under null model $\mathcal{M}_0$ and full model $\mathcal{M}_1$; and, respectively, p_0 is the probability of (true) null model. Currently, p_0 is temporarily considered as a known constant factor. How to estimate p_0 still needs further research.

The final question is about how to estimate σ_i. We can estimate σ_i from the sets of log-fold-changes (LFCs) from fitting BS subsamples using the standard ***edgeR*** pipeline:

$$\hat{\sigma}_i^2 = var(C_i), \tag{37}$$

where $C_i = \{LFC_{ij}^{[i \bullet 1]}, LFC_{ij}^{[i \bullet 2]}, \ldots, LFC_{ij}^{[i \bullet m]}\}$. Each element, $LFC_{ij}^{[i \bullet z]}$, is the estimated LFC of the parameter interest j (i.e., the jth coefficient to be tested equal to zero in the model) using the standard ***edgeR*** pipeline to fit a BS subsample.

References

[1] William H Aeberhard, Eva Cantoni, and Stephane Heritier. Robust inference in the negative binomial regression model with an application to falls data. *Biometrics*, 70(4):920–931, 2014.

[2] Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Genome biology*, 11(10):R106, jan 2010.

[3] Douglas Bates, Martin Mächler, Ben Bolker, and Walker. Fitting Linear Mixed-Effects Models using lme4. *Cran Vignette*, pages 1–51, 2014.

[4] C S Berkey, D C Hoaglin, F Mosteller, and G A Colditz. A random-effects regression model for meta-analysis. *Stat Med*, 14(4):395–411, 1995.

[5] L Breiman, J H Friedman, R A Olshen, and C J Stone. *Classification and Regression Trees*, volume 19. 1984.

[6] Peter Bûhlmann. Bagging, Subagging and Bragging for Improving some Prediction Algorithms. In *Recent Advances and Trends in Nonparametric Statistics*, pages 19–34. Elsevier Inc., 2003.

[7] D R Cox. *Principles of Statistical Inference*. Cambridge University Press, 2006.

[8] Robert Delongchamp, Taewon Lee, and Cruz Velasco. A method for computing the overall statistical significance of a treatment effect among a group of genes. *BMC bioinformatics*, 7 Suppl 2(Suppl 2):S11, 2006.

[9] Bradley Efron, Robert Tibshirani, John D Storey, and Virginia Tusher. Empirical Bayes Analysis of a Microarray Experiment. *Journal of the American Statistical Association*, 96(456):1151–1160, 2001.

[10] John Fox and Sanford Weisberg. Robust Regression in R. *An R Companion to Applied Regression*, (December):1–17, 2012.

[11] Ulrich Halekoh and Søren Højsgaard. A Kenward-Roger Approximation and Parametric Bootstrap Methods for Tests in Linear Mixed Models–The R Package pbkrtest. *Journal of Statistical Software*, 59(1):1–32, 2014.

[12] Larry V. Hedges and Ingram Olkin. *Statistical methods for meta-analysis*, volume 72 of *Plant-Insect Interactions*. Academic Press New York, 1985.

[13] J L Jensen. A Historical Sketch and Some New Results on the Improved Log Likelihood Ratio Statistic. *Scandinavian Journal of Statistics*, 20(1):1–15, 1993.

[14] Guido Knapp and Joachim Hartung. Improved tests for a random effects meta-regression with a single covariate. *Statistics in Medicine*, 22(17):2693–2710, 2003.

[15] Charity W Law, Yunshun Chen, Wei Shi, and Gordon K Smyth. Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biology*, 15(2):R29, 2014.

[16] Jeffrey B. Lewis and Drew A. Linzer. Estimating regression models in which the dependent variable is based on estimates. *Political Analysis*, 13(4):345–364, 2005.

[17] Steven P Lund, Dan Nettleton, Davis J McCarthy, and Gordon K Smyth. Detecting Differential Expression in RNA-sequence Data Using Quasi-likelihood with Shrunken Dispersion Estimates. *Statistical Applications in Genetics and Molecular Biology*, 11(5), 2012.

[18] Marianthi Markatou. Robust statistical inference: weighted likelihoods or usual M-estimation? *Communications in Statistics–Theory and Methods*, 25(11):2597–2613, 1996.

[19] Marianthi Markatou, Ayanedranath Basu, and Bruce Lindsay. Weighted likelihood estimating equations: The discrete case with applications to logistic regression. *Journal of Statistical Planning and Inference*, 57(2):215–232, 1997.

[20] Marianthi Markatou, Ayanendranath Basu, and Bruce G Lindsay. Weighted likelihood equations with bootstrap root search. *Journal of the American Statistical Association*, 93(442):740–750, 1998.

[21] Harold J Pimentel, Nicolas Bray, Suzette Puente, Páll Melsted, and Lior Pachter. Differential analysis of RNA-Seq incorporating quantification uncertainty. *bioRxiv*, 2016.

[22] Jose Pinheiro, Douglas Bates, Saikat DebRoy, and Deepayan Sarkar. *nlme: Linear and Nonlinear Mixed Effects Models*. 2007.

[23] Stephen W Raudenbush. Analyzing effect sizes: Random-effects models. *The handbook of research synthesis and meta-analysis*, 2:295–316, 2009.

[24] Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 71(2):319–392, 2009.

[25] Frank L Schmidt and John E Hunter. *Methods of meta-analysis: Correcting error and bias in research findings*. Sage publications, 2014.

[26] H Skaug, D Fournier, A Nielsen, A Magnusson, and B Bolker. Generalized linear mixed models using AD model builder. *R package version 0.7*, 2, 2012.

[27] Gordon K Smyth. Limma : Linear Models for Microarray Data. *Bioinformatics*, pages(2005):397–420, 2005.